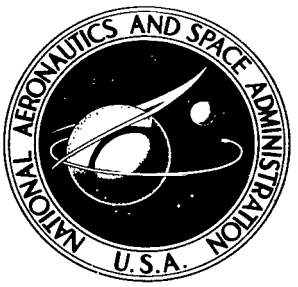
NASA TECHNICAL NOTE

NASA TN D-8328



NASA TN D-8328

A REAL-TIME, DUAL PROCESSOR SIMULATION OF THE ROTOR SYSTEM RESEARCH AIRCRAFT

D. Brian Mackie and Thomas S. Alderete

Ames Research Center
Moffett Field, Calif. 94035

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • JANUARY 1977

1. Report No. NASA TN D - 8328	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle A REAL-TIME, DUAL PROCESSOR SIMULATION OF THE ROTOR SYSTEM RESEARCH AIRCRAFT		5. Report Date January 1977 6. Performing Organization Code
7. Author(s) D. Brian Mackie and Thomas S. Alderete*		8. Performing Organization Report No. A-6652 10. Work Unit No. 504–09–42
9. Performing Organization Name and Address Ames Research Center Moffett Field, Calif. 94035 and *Computer Sciences Corporation Mountain View, Calif. 94040		11. Contract or Grant No. 13. Type of Report and Period Covered Technical Note
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20456		14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

A real-time, man-in-the loop, simulation of the Rotor System Research Aircraft (RSRA) was conducted at Ames Research Center. The unique feature of this simulation was that two digital computers were used in parallel to solve the equations of the RSRA mathematical model. This paper documents the design, development, and implementation of the simulation. Program validation is discussed and examples of data recordings are given. This simulation provided an important research tool for the RSRA project in terms of safe and cost-effective design analysis. In addition, valuable knowledge concerning parallel processing and a powerful simulation hardware and software system has been gained by the Ames flight simulation laboratory.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Simulation Real-time Dual-processor Rotor System Research Aircraft	Unlimited STAR Category – 05

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	73	$4.25

* For sale by the National Technical Information Service, Springfield, Virginia 22161

TABLE OF CONTENTS

A REAL-TIME, DUAL PROCESSOR SIMULATION OF THE ROTOR SYSTEM RESEARCH AIRCRAFT

D. Brian Mackie
Ames Research Center

and

Thomas S. Alderete
Computer Sciences Corporation

SUMMARY

A real-time, man-in-the loop, simulation of the Rotor System Research Aircraft (RSRA) was conducted at Ames Research Center. The unique feature of this simulation was that two digital computers were used in parallel to solve the equations of the RSRA mathematical model. This paper documents the design, development, and implementation of the simulation. Program validation is discussed and examples of data recordings are given. This simulation provided an important research tool for the RSRA project in terms of safe and cost-effective design analysis. In addition, valuable knowledge concerning parallel processing and a powerful simulation hardware and software system has been gained by the Ames flight simulation laboratory.

INTRODUCTION

A real-time, man-in-the-loop, flight simulation of the Rotor System Research Aircraft (RSRA) was conducted at the Ames Research Center. The large-scale simulation was implemented on the Flight Simulator for Advanced Aircraft (FSAA) using the hybrid computation facilities of the Flight and Guidance Simulation Laboratory. This research was conducted as part of a joint NASA-Army project to develop the RSRA.

The unique feature of this simulation was that two digital computers were used in parallel to solve the equations of the RSRA mathematical model. This dual-processor configuration was necessary to obtain acceptable cycle times for a real-time simulation of the complex RSRA math model.

This paper documents the hardware and software aspects of the RSRA simulation. Although the physical characteristics of the aircraft and of the mathematical model are described, the complete mathematical model should be consulted for details.[1] The FSAA facilities are also discussed, but again the interested reader should see reference 1 for further detail. The standard software used in the FSAA facility is described in general and the simulation programs for RSRA are explained. Implementation of the simulation on the dual processor system is discussed, including the loop structure, data transfers, and timing. Finally, in the appendices, the simulation

[1] Howlett, J. J.: RSRA Simulation Model. Volume 1 and Volume 2. SER 72009, Contract NAS1-13000, Sikorsky Aircraft. October, 1974.

programs are flowcharted, the FSAA cockpit layout is described, the COMMON lists and transfer lists are given, and some samples of data are shown.

The special techniques derived for this simulation constitute a practicable method for alleviating the real-time constraint. The system that was developed has since been used for the simulation of a tilt rotor aircraft.

ROTOR SYSTEM RESEARCH AIRCRAFT SIMULATION

Aircraft Description

The Rotor System Research Aircraft (RSRA) is a versatile research vehicle that will provide a flying test bed capability for experiments and tests of new rotor concepts. Rotor systems, such as variable geometry rotors, variable diameter rotors, reverse velocity systems, as well as more conventional rotor systems, will be accommodated by the RSRA. The aircraft has three flight configurations: (1) helicopter mode, (2) fixed-wing mode, and (3) compound mode. In the most complex configuration, the compound mode, the RSRA is fitted with a main rotor, a tail rotor, wings, a vertical stabilizer, two horizontal stabilizers, two jet propulsion engines, and a full complement of both rotary-wing and fixed-wing controls (see fig. 1). In this configuration, the RSRA is capable of approximately 155 m/s (300 knots) with a gross weight of 115,700 N (26,000 lb).

There are plans to build two aircraft, each of a different configuration. Both will be instrumented to measure aerodynamic loads while in flight. In addition, a variable on-board ballast system will make it possible to vary the center of gravity.

The two aircraft will be supplied with Sikorsky's S-61 rotor systems. This is a 5-bladed, 19-m (62-ft) diameter rotor. A conventional 5-bladed, 3.23-m (10.6-ft) diameter tail rotor will provide main rotor torque balance. The wing has a 12.8-m (42-ft) span and an area of approximately 34.37 m^2 (370 ft^2). The empennage is an I-tail configuration. The wing and lower horizontal stabilizer have coupled, variable incidence. The propulsion engines are General Electric TF34-GS-2 turbofans with a static thrust rating of 36,308 N (8159 lb). The engines providing main rotor power are two General Electric T58-GS-5 turboshafts with a static rating of 1044 shaft kw (1400 shaft horsepower). The control system is a complex combination of helicopter and fixed-wing aircraft controls. In addition to a full set of controls for such individual items as collective pitch and flaps, there are mixing and phasing functions with control phasing being a manual operation. There is a full set of rotary-wing and fixed-wing series trims. The series trim system is designed to compensate control setting changes due to control phasing inputs as well as providing pilot trim changes. There is an analog stability augmentation system (SAS) providing rate gyro stabilization, and an augmented force feel system (FFS) to provide variable stick-feel characteristics.

Another aspect of the control system (one not simulated at Ames) is the Electronic Flight Control System (EFCS). The EFCS, which includes an on-board digital computer, provides a fly-by-wire capability which can be overridden by a mechanical control system. The EFCS allows automatic flights for data acquisition runs and pre-programmed tests.

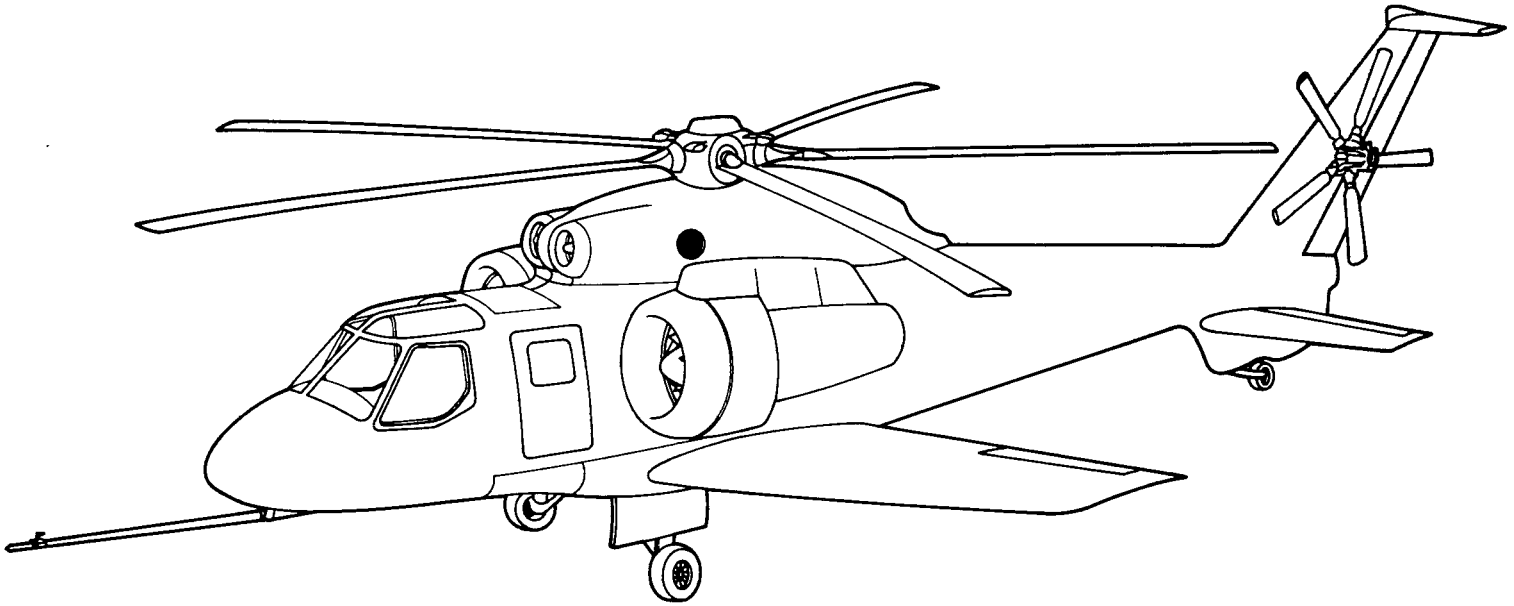


Figure 1.– Rotor System Research Aircraft.

Discussion of Mathematical Model

The RSRA mathematical model is a non-linear, highly coupled, large angle, rigid body formulation in six degrees of freedom with three additional degrees of freedom describing rotor dynamics. The model evolved from a general helicopter simulation model developed by Sikorsky for helicopter design and analysis. The basic aerodynamic data for the airframe was derived from the Sikorsky 1/6-scale-model wind tunnel experiments.

Each aircraft subsystem (e.g., tail, engine) has been modeled separately with the equations grouped in a modular fashion. The equations for each subsystem are coupled by aerodynamic interference effects and, in the case of the rotor, by inertial effects. This modular design allows straightforward modification to airframe configuration, identification of subsystem performance, and good representation of complex interference phenomena. The mathematical model allows for all the RSRA flight configurations: helicopter, fixed-wing, and compound. There is also provision for an in-flight configuration change – blade severance – activated by the main rotor blade release switch.

The rotor subsystem is the most complex subsystem of the model; solution of the equations requires more computation time than does the rest of the simulation. The development of the equations is based on the blade element theory with the following basic assumptions.

1. The rotor blades are considered rigid.
2. Air mass flow through the rotor disc is represented by applying a simple lag to downwash.
3. A uniform mass distribution is assumed over the span of each rotor blade.
4. The aerodynamic effects of yawed flow on a blade element can be derived from the unyawed flow data using simple sweep theory.

Each blade is segmented into spanwise elements with airflow being considered uniform over each segment. Figure 2 illustrates the blade segments and a typical lead-lag blade hinge model. The number of segments is a program variable and the segments can either be specified at equidistant radial positions or at radial positions such that all segments sweep out equal area annuli. Using the angle of attack and local velocity vector, lift and drag forces are computed for every blade element and are summed for each blade. The mass and inertial loads (functions of blade and airframe motion) are added to the aerodynamic loads to give the total shear at the blade root hinge pins. The total shear forces are then summed around the azimuth. Rotor moments are computed from the hinge offsets. Rotor blade flapping and lagging degrees of freedom are computed from the

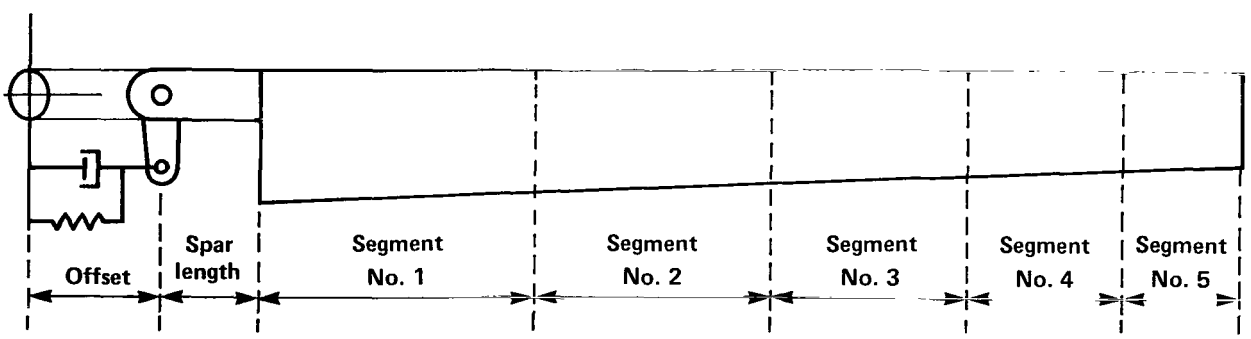


Figure 2.– Blade element definition.

aerodynamic and inertial moments at the blade hinges. Referring again to figure 2, a typical lead-lag hinge is shown to be represented by a spring and a damper.

Many special considerations have been given to the rotor model. The equations have been optimized for real-time computation by extensive parameter groupings. Special integration schemes are used which make use of the Fourier representation of flapping and lagging. The rotor forces and moments that are passed on to the airframe undergo filtering using Kalman filter techniques. This filtering is necessary in some circumstances to suppress the amplitude of artificial rotor oscillations which result from a reduction in the number of simulated blades. Finally, there is provision for the real-time release of all rotor blades.

The fuselage, wing, and nacelle aerodynamics have been modeled together for direct correlation with wind tunnel data. The equations for local velocities and angles of attack include rotor downwash effects. The wind axis forces and moments are given in tables as functions of angles, control surface deflections, and power settings. The total forces and moments are transformed into the body axis system. Various airframe configurations are accommodated by switching from one set of tables to another. In this simulation, two sets of aerodynamic function tables were used: one for a fuselage with wing and nacelles and one for the fuselage alone.

The empennage has been modeled as a separate subsystem. This allows for modifications to tail configuration without altering other aerodynamic subsystems. The configuration modeled for this simulation is an "I" configuration with an optional flying tail design for the lower horizontal stabilizer. The local velocities and angles of attack take account of the rotor plus airframe wash and the propulsion engine efflux. The local dynamic pressure loss is likewise influenced by airframe interference effects. The aerodynamic forces and moments are determined from function tables. Provision is made for tail configuration change, because in helicopter mode the tail is a "T" design.

The tail rotor model, which is quite simple, uses the linearized, closed form Bailey theory. The Bailey theory relates forces and moments to three parameters: inflow ratio, advance ratio, and blade pitch angle. Simplifying assumptions limit the theory at high velocities where compressibility effects are significant and at high blade angles where stall effects are evident. The total thrust vector is computed, then transformed into body axis forces. The local velocities imposed on the tail rotor are found in the same manner as for the empennage. The thrust vector is modified due to flow blockage by the vertical fin.

The propulsion engines are modeled by matching performance characteristics with static and dynamic performance data. The use of a variable pure time delay plus a variable first order lag in thrust response is the basis of the modeling. The output of this subsystem is the direct thrust and the air mass flow rate, nacelle aerodynamic effect being a part of the fuselage-wing-nacelle subsystem.

The final major subsystem is the control system. The modeled control system includes all components necessary to evaluate the aircraft's flying qualities, including the Stability Augmentation System (SAS) and the Force Feel System (FFS). The control system itself is represented by gearing, mixing, and phasing with no actuator or linkage dynamics. The SAS is a conventional rate gyro stabilization system, expressed as transfer functions in Laplace notation. The FFS characteristics are specified in the model, but implemented on a programmable McFadden loader. The model has a complete set of both rotary-wing and fixed-wing primary controls; that is, lateral and

longitudinal cyclic pitch, collective pitch for both main rotor and tail rotor, ailerons, elevator, and rudder. The control power of the primary controls (phasing between helicopter and fixed-wing) is determined by the positions of the three Control Phasing Unit (CPU) levers. The secondary controls include flaps, wing incidence, and drag brake. The rotor engine power is controlled by an overhead lever and the fan engine power is controlled by twist grips on the collective stick. There are series trim beeper systems for each primary helicopter and fixed-wing control plus beepers for the engines and CPU's. There are proportional parallel trims for the FFS. In addition, there is a CPU null system which nulls the control position shift due to CPU lever motion. This nulling occurs through the series trim actuators.

Most of the above controls, plus the instrument panel, can be seen in the picture of the RSRA simulation cockpit in figure 3.

Simulation of the RSRA

Consideration of the real-time simulation of the RSRA math model can be divided into three categories: (1) simulation hardware, (2) simulation software, and (3) implementation of the math model. Simulation hardware is the collection of physical equipment necessary to perform the simulation. This includes such elements as computers, data recording devices, cockpit and motion system, and visual scene generators. Simulation software is the collection of programs within the digital computer or computers. The main purpose of these programs is to accept input commands from the pilot or signal generating programs within the computer, perform computations on these signals based on the math model, and output control commands to the hardware based on these computations. Implementation of the math model refers to the specific action taken to integrate the aircraft specification into the given hardware and software structure.

Simulation Hardware

The hardware elements used to perform a flight simulation at Ames Research Center are collectively referred to as the Flight Simulator for Advanced Aircraft (FSAA). A simplified FSAA system diagram for the RSRA is shown in figure 4. The specific configuration of the FSAA system shown in figure 4 is referred to as the INTERTIE system.

A Xerox Data Systems (XDS) Sigma 7 and an XDS Sigma 8 are used to provide control values for the hardware elements shown in the diagram, and to process feedback signals from these elements. The two digital computers communicate with each other via the XDS 7908 interface hardware attached to the Multiplexed Input-Output Processor (MIOP) on each computer. The computers communicate with the outside world via analog to digital converters (ADC's), digital to analog converters (DAC's), and discrete input/output (DIO) channels. Necessary buffering and patching are performed by three analog computers.

The simulation engineers and the researchers conduct the experiments from the Simulation Control Station. This station consists of two CRT's with keyboards: one for communication with the software debug system on the Sigma 8, and the other for the Sigma 7. An integral part of this control station is a series of programmable pushbuttons and light indicators, one group for the

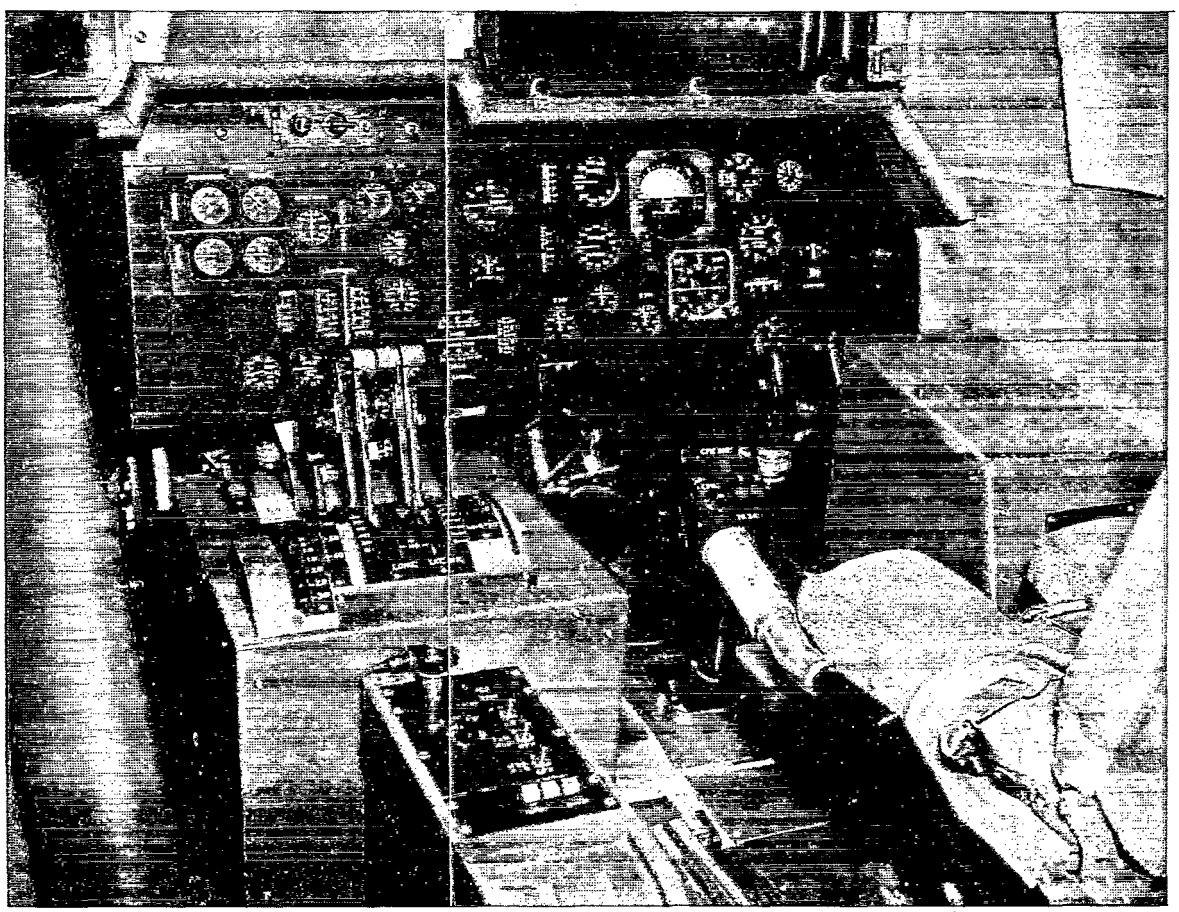


Figure 3.— RSRA simulation cockpit layout.

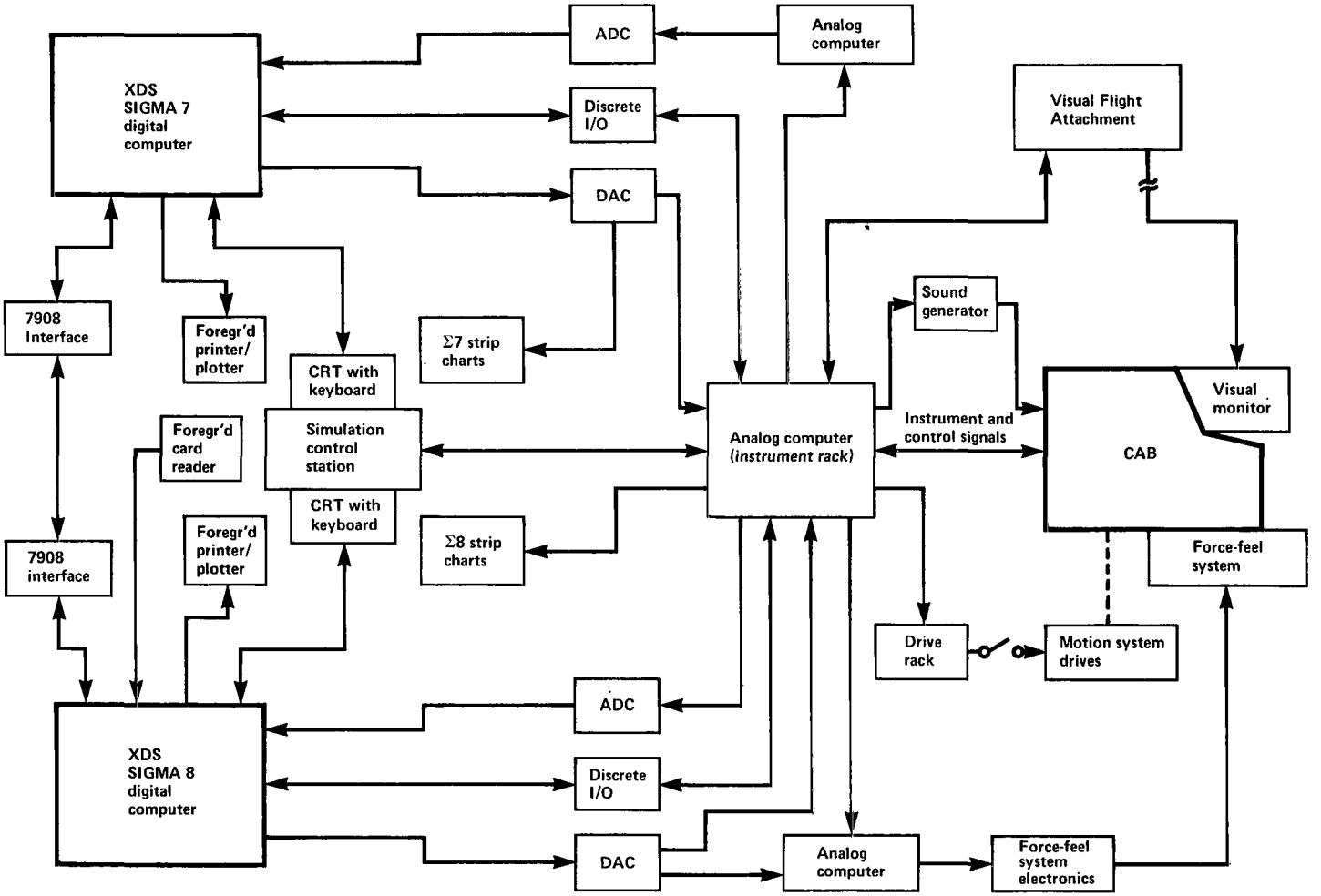


Figure 4.– INTERTIE System.

simulation engineers and another for the researchers. The control station also includes two electrostatic printer/plotters, six strip chart recorders, and a foreground card reader.

The majority of the communication from the digital and analog computations to the hardware related to the moving base simulator is patched through an analog computer which serves as an instrument rack. From the instrument rack, signals are sent to the Visual Flight Attachment which generates the pilot's visual scene. This system consists of a mobile color television camera and a three dimensional terrain model, which allows the representation of a scene up to 3.4 miles wide by 13 miles long. Signals are also sent to the sound generator for aural cues, and to the drive rack, from which signals are sent to the motion drive system for acceleration cues. The motion drive system consists of a complex movable structure that can drive the cockpit in a combination of six directions at once: three translational (lateral ±50 ft, longitudinal ±4 ft, vertical ±5 ft) and three rotational (roll ±45°, pitch ±22-1/2°, yaw ±30°).

Signals are also sent to an analog computer to provide values for the force-feel system (FFS) electronics. Computations by the FFS electronics produce values for the electro-hydraulic FFS, providing the pilot with forces in the lateral and longitudinal stick and in the rudder pedals which correspond with those encountered in actual flight. Thus, gradient, break-out, friction, damping, detent, deadband, bias, hysteresis, and variable stops can all be provided. Finally, drive signals are sent to the instruments and received from the controls in the cab. Further specific information on the configuration of the controls and instruments for the RSRA simulation is given in appendix A. Those interested in a further discussion of the FSAA system may refer to reference 1.

Simulation Software

The collection of computer programs necessary to perform the simulation may be considered in three main groups: (1) Systems Programs and Function Table Processor, (2) BASIC subroutines, and (3) user-supplied subroutines.

Systems Programs and Function Table Processor.– There are three systems programs used by all simulations on the FSAA. These three programs are FATHER, MOTHER, and CASPRE; they are known collectively as FAMILY III.

The first of these, FATHER (Foreground Area Time Handling Executive Routine), is the real-time executive. FATHER performs the following major tasks:

1. Loads the simulation into the proper area in memory
2. Processes all user-issued commands
3. Controls input/output
4. Processes all traps and FORTRAN run-time errors

The second systems program, MOTHER (Monitor Time Handling Executive Routine), is the real-time scheduler. MOTHER performs the following major tasks:

1. Builds fixed set of lists during initialization which schedule BASIC and user supplied subroutines during operation
2. Schedules input/output

3. Processes mode requests
4. Outputs error messages related to scheduling and mode requests

The third systems program, CASPRE (Computer Aids to Simulation Programmers and Engineers), consists of two internal modes: Executive CASPRE and Debug CASPRE. Executive CASPRE performs the following major functions:

1. Loads FATHER, MOTHER, and user programs
2. Activates Debug CASPRE
3. Saves and restores program data upon request
4. Terminates the user from the foreground area

Debug CASPRE performs the following major functions:

1. Allows the user to read the contents of any memory location
2. Allows the user to change any data contained in his area of memory
3. Allows breakpoints to be set
4. Performs memory searches upon request
5. Allows entire memory blocks to be output to a user-specified device

For further information on FATHER, MOTHER, and CASPRE see reference 2.

The INTERTIE software is a modified version of the FAMILY III software system. Both FATHER and MOTHER are modified to handle the dual processes and the inter-computer communications. CASPRE is essentially unaltered and a totally independent CASPRE exists on each machine. The Sigma 8 is the master controller and performs scheduling, handles mode requests, controls data transfers between computers, and triggers the Sigma 7 clock.

The modes are the same as in FAMILY III; the essential ones being I.C., HOLD, and OPERATE. In I.C. (initial condition), the processes are synchronous, but sequential, that is, only one process on one computer is being executed at a given time. In HOLD, the processes are halted, holding output values constant. In OPERATE, the processes are synchronous, interruptible, and running simultaneously in real-time on both computers.

The data transfers are controlled by a sequence of interrupts and protocol words. The protocol words contain information about status, mode, and the data transfer. The reading and writing of protocol words are triggered by the interconnected interrupts. Data is sent via the 7908 Interface and may originate within the system software or by request from the user program. The user program can send and receive data between computers via FORTRAN callable subroutines.

The systems software includes a special purpose compiler called the Function Table Processor (FTP). Much of the data in a simulation math model may be in the form of tables with one or more independent variables. The FTP accepts user written programs which list this tabulated data. It processes this data off-line in non-real-time and outputs an object program with control words and specially formatted data tables. This object program is then used to provide a fast, efficient linear-interpolation process for table look-ups during the real-time operation mode.

BASIC Subroutines.– BASIC is a collection of subroutines that are basic or common to most flight simulations. BASIC is intended to avoid duplication of effort with each new simulation, and also to allow standardization of simulations in order to simplify simulation checkout. The subroutines of the BASIC system are designed to:

1. Provide a solution to the equations of motion of a rigid-body vehicle
2. Generate signals to drive hardware devices that provide the pilot with motion, visual, and aural cues
3. Provide for evaluation of dynamic vehicle response in real-time or time scaled mode without pilot or cab on-line
4. Provide for extensive real-time and non-real-time data acquisition and manipulation capabilities
5. Provide for common sets of variables and discretes for standardized communication among BASIC subroutines, with user subroutines and with the "outside world."

More extensive information on the BASIC system is given in reference 3.

User-Supplied Subroutines.– The user-supplied subroutines are written by the simulation engineer and are based on the math model supplied by the researchers. They communicate among each other and with the BASIC subroutines via COMMON blocks. Copies of the COMMON blocks developed for RSRA are given in appendix B.

The user-supplied subroutines describe the static and dynamic characteristics of the particular aircraft being simulated. Among the quantities produced by these subroutines, most important are the six-component aerodynamic and propulsive forces and moments, which correspond to the aircraft's six degrees of freedom. These forces and moments are communicated via the BASIC COMMON blocks to the BASIC subroutines, which use these values to calculate accelerations, velocities, positions, and other important quantities discussed above. The major user subroutines obtained from the math model for RSRA include computations of forces, moments, and other important quantities for the rotor, wing-fuselage-nacelles, controls, engines, tail, and tail rotor. These subroutines correspond to aircraft subsystems defined by the mathematical model. Included in these subroutines is the capability to initiate failures in several controls and control surfaces. Each of these subroutines is described in greater detail in appendix C. Other user-supplied subroutines include a linear atmospheric model, a trim subroutine, instrument drives, data tables, print routines, and controllers for transfer of data between the Sigma 7 and Sigma 8.

Implementation of the Mathematical Model

There are two phases in representing the math model in the form of a computer program. First, the subroutines must be written to solve the equations of the model. Secondly, the subroutines must be arranged in the proper order; that is, the discrete representation must be dynamically similar to the continuous process.

As noted earlier, the computer subroutines for the RSRA math model are described in appendix C. For each subroutine, there is a brief description, and a flow chart that defines the computations performed by the subroutine.

The second phase in implementation of the math model, ordering of the subroutines, is described below. This ordering of the subroutines is referred to as the simulation "loop structure." Since information transfer is an inseparable part of the loop structure in a dual processor configuration, it is also discussed below.

The subroutines described in appendix C and the BASIC subroutines necessary for the simulation are configured on the Xerox Sigma 7 and Sigma 8 as follows. On the first pass through both computers, main subroutines on each computer set up frame times and make a series of calls to subroutines in MOTHER. As described earlier, MOTHER then creates tables which control program flow from this point on. Each time the simulation is commanded to begin execution, either for the first time or after stopping for any reason, this procedure is repeated, allowing the simulation engineer to change frame times on line.

A conceptual diagram of the loop structures for the Sigma 7 and Sigma 8 is shown in figure 5. The Sigma 7 is primarily responsible for computation of the fixed-wing aircraft forces and moments, and the Sigma 8 is primarily responsible for computation of the main rotor forces and moments.

Aircraft forces and moments on the Sigma 7 result from forces and moments due to the following subsystems: (1) wing, fuselage, and nacelles; (2) propulsion engines; (3) tail; and (4) tail rotor. Aircraft forces are added to main rotor forces and the results are used to compute linear position information. Aircraft moments are transmitted to the Sigma 8.

Main rotor forces and moments on the Sigma 8 result from computations based on main rotor control positions. Main rotor moments are added to aircraft moments and these results are used to compute body rotational information. Main rotor forces are transmitted to the Sigma 7. Also resident on the Sigma 8 are computations for motion and visual drives.

The loop structure is amplified further in figure 6 to give a clearer view of how the computations and the information transfers are linked. A list of the elements of the data transfer lists can be found in appendix D.

Referring to figure 6 and appendix D, the following major factors become evident. List No. 1 consists primarily of aircraft rotational information and failure indicators. The aircraft rotational information (roll, pitch, and yaw positions, rates, and accelerations) is computed near the end of each pass through the Sigma 8 and, through the transfer list, this information is introduced almost immediately back into the fixed-wing aircraft calculations on the Sigma 7. The failure indicators had to be transferred because the failure switches were "hardwired" into the Sigma 8 research panel, whereas most of the elements to be failed reside on the Sigma 7.

List No. 2 consists of values computed by the rotor subroutine on the Sigma 8. Rotor downwash is used extensively in aerodynamic calculations for the fixed-wing aircraft. Rotor translational forces are added to fixed-wing forces and used in the calculation of total aircraft translational information on the Sigma 7.

List No. 3 consists of body translational velocities and accelerations and of force-feel system values. The body axis velocities and accelerations are used in the rotor computations to compute blade segment velocities and accelerations. Force-feel system values are computed on the Sigma 7 in

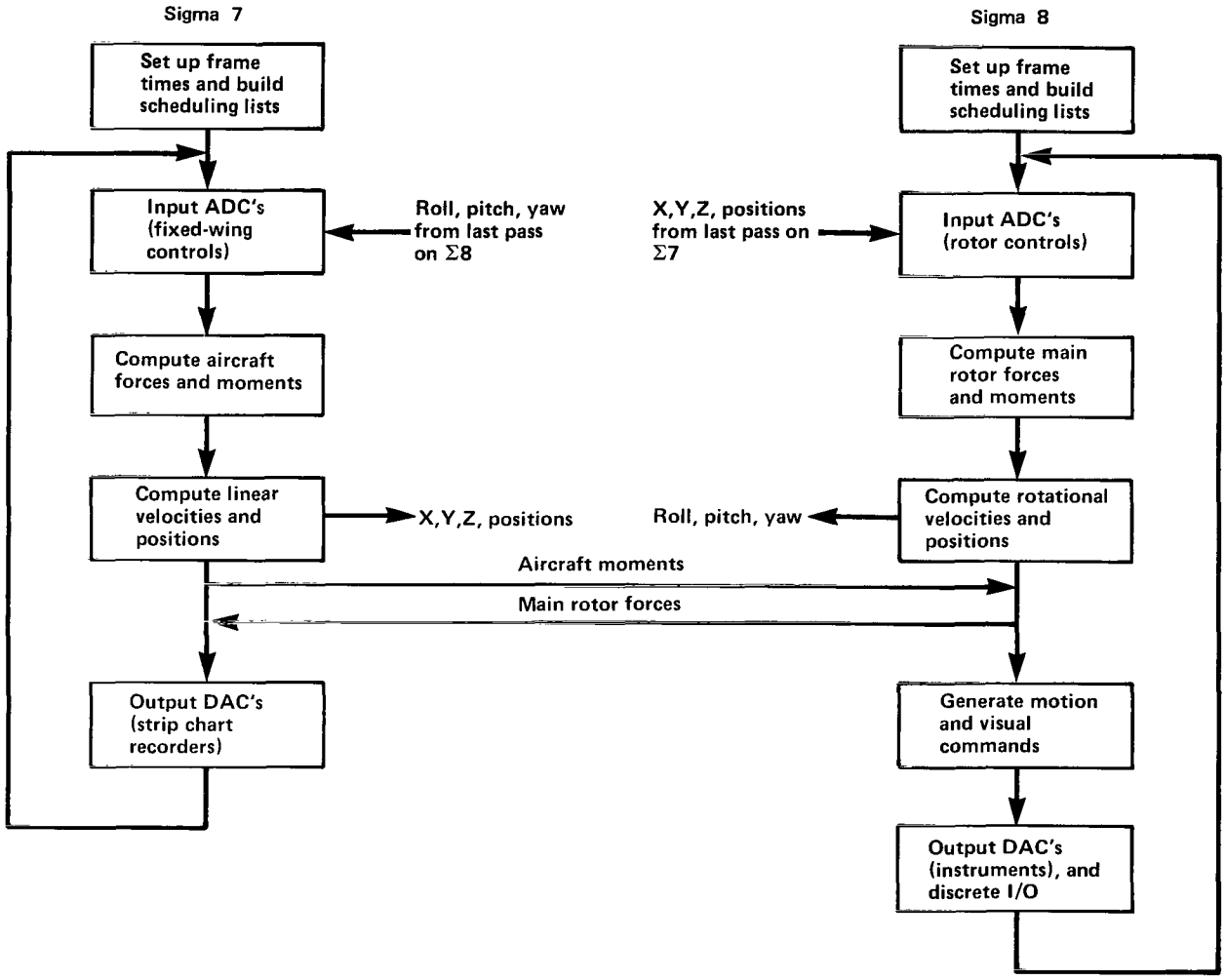


Figure 5.– Conceptual INTERTIE loop structure.

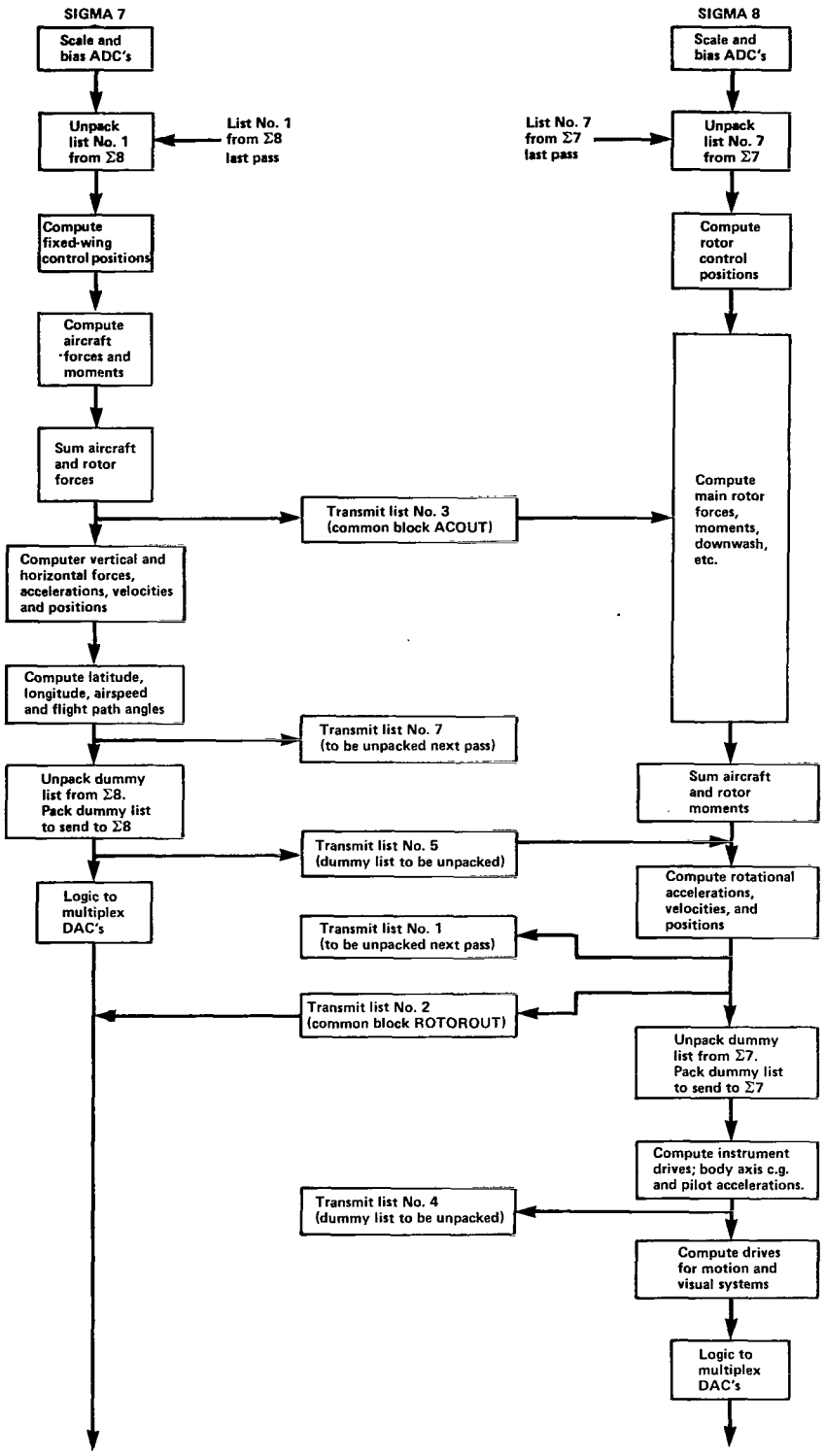


Figure 6.— Linkage of computations and information transfers.

an attempt to maintain a balance in the division of labor between the two computers. It is necessary to transfer these values to the Sigma 8 because the force-feel system is driven from the Sigma 8.

Lists No. 4 and No. 5 are "dummy" lists. These lists allow the simulation engineer to patch in transfers, which may not have been foreseen at program time, during actual operation of the simulation. This eliminates the necessity of stopping to program new transfers during operation time.

List No. 7 consists of aircraft translational information and of other values calculated on the Sigma 7. Also, note near the end of the items in list No. 7, the roll, pitch, and yaw moments due to the fixed-wing aircraft. After the computation of rotor roll, pitch, and yaw moments on the Sigma 8 (which vary greatly in value from one frame to the next) these moments are added together and used immediately to compute entire aircraft rotational information.

Finally, there are two transfer lists, No. 6 and No. 8, which are not shown in figure 6. These lists are transferred between the two computers in I.C. mode only. Values on these lists supply important initial conditions that are not needed in operate mode. Some of these values are transferred to eliminate the necessity of entering changes on both keyboards at the simulation control station.

To further explain the organization of the INTERTIE loop structure, some examples can be cited. Two critical areas are rotor stability and pilot cues. Due to flapping degrees of freedom in the rotor model, the rotor subroutine produces high frequency moments. As a result, instabilities will occur in the dual computer simulation which necessitate the inclusion of two subroutines on the Sigma 8. The first subroutine, TORQ8, is a simple subroutine which accepts total fixed-wing-aircraft moments from the Sigma 7, and rotor moments from the Sigma 8, and adds them together to produce total aircraft moments. The fixed-wing-aircraft moments remain relatively stable, so the slight delay as a result of their transfer from the Sigma 7 is not critical.

The second subroutine, BROTATE, utilizes total aircraft moments, plus other information, to compute total aircraft rotational degrees of freedom; that is, rotational accelerations, velocities, and displacements. The inclusion of this subroutine plus TORQ8 on the Sigma 8 eliminates high frequency instability, especially in roll acceleration.

Immediate rotor control response to pilot control inputs in crucial in the evaluation of rotary-wing craft. Therefore, it is necessary to include on the Sigma 8, a subroutine that will process control inputs related to main rotor control. The inclusion of this subroutine immediately preceding the rotor subroutine eliminates unacceptable delays in rotor response to control inputs.

Many of the decisions made by the pilot of a rotor-craft result from the responses of the pilot to motions of the aircraft. Thus, the pilot will often make control responses to accelerations which he feels, even before his instruments or visual scene indicate changes in flight parameters. Because of this, it is necessary to include another subroutine, BMOTION, on the Sigma 8. This subroutine computes the values that are used to drive the motion system on the FSAA. The inclusion of this subroutine immediately following the rotor subroutine allows a more closely representative motion response to high frequency aircraft accelerations.

Because the motion-driving subroutine has been moved to the Sigma 8, it is essential also to move to the Sigma 8 the visual-driving subroutine, BVISUAL. Otherwise, the pilot would be confused when the visual scene lags behind motion that he feels from the motion system.

Special Considerations

It is a continuing effort at the Ames Flight Simulation Laboratory to standardize simulation methods for the purpose of making new programs easier to create. The RSRA presented exceptions to the standard methods and several areas of the simulation required special attention.

Information Transfer

One of the major considerations with this simulation is information sharing between the two computers. The processes on each machine are not independent of each other; that is, there are computations on the Sigma 8 that require data resulting from computations on the Sigma 7, and vice versa. The objectives in this information sharing are to transfer the complete set of data required, to transfer the data so that proper computational order will be maintained, and to make all transfers in as short a time as possible.

There are three distinct ways in which the data transfers are handled, all utilizing the INTERTIE software. The first method is to construct a table of variable names. This is required when the data to be transferred are scattered throughout existing data structures. The second method is to transfer a block of contiguous variables. This is useful when an entire COMMON block is to be transferred. In the third method, a block transfer is used, but the user program conditionally packs and unpacks the buffer. This method is essential in cases where some variable is required on both computers and can be altered on either computer. Without some logic to unpack such a variable, its value will oscillate between the two computers.

In the RSRA simulation, an effort was made to optimize the transfer times and to insure proper computational order by correct placement of the transfers. These two objectives, timing and placement of transfers, are interrelated because transfers can occur simultaneously with computations. The objectives are further complicated (or controllable, depending on viewpoint) by the fact that list transfers are distributed on the receiving computer only on request, but block transfers are distributed at the time of receipt. Major considerations in information transfers were as follows: (1) a transfer should be made as soon as all the variables in a list become available; (2) a transfer (block) should not occur before the receiving computer is computationally ready to receive; (3) bunching together of transfers should be avoided; and (4) the number of individual transfers should be minimized.

There were some additional considerations for the data communications. Some variables only had to be sent during I.C. mode; most of these variables were BASIC initial conditions. Since I.C. mode is a sequential non-real time process, timing and order of the I.C. data transfers were not a problem. One further consideration was that no transfers could occur on either machine until that particular transfer list had been initialized on both computers. This was a problem only during the first time through the program.

Timing

The real-time cycle rate was a considerable problem with the RSRA simulation. The integrity of the entire simulation, especially the rotor model, was heavily dependent on program cycle time. The program cycle time is the interval of time between communications with the simulation hardware, and is also the interval of integration in all solutions of differential equations. Table 1 shows a breakdown of the total program cycle time by functional blocks.

The actual computer cycle time is the greater of the times required on each computer. Thus, the times shown for the Sigma 8 are the true cycle times. Notice that the cycle time is dependent on configuration (helicopter or compound) and is stated for one rotor condition; four blades, three segments.

It is interesting to note the advantage in cycle time using two computers. Two factors must be considered when examining the cycle time: (1) the Sigma 8 is approximately 30 percent faster than the Sigma 7, and (2) when using one computer the "overhead" of data transfers is omitted. To establish a point of reference, some hypothetical frame times are given.

1. Assuming one Sigma 8, the frame time would be 64 msec.
2. Assuming two Sigma 8's in the INTERTIE configuration, with an equal division of computational load, the frame time would be 40 msec.
3. Assuming one Sigma 8 and one Sigma 7, the INTERTIE configuration, and an equal division of labor, the cycle time would be 45 msec.

From table 1, the actual frame time for RSRA on INTERTIE was 46 msec. Comparing this frame time with the examples above, it is seen that the frame time would have increased by 40 percent (from 46 to 64 msec) if one Sigma 8 (no INTERTIE) had been used. Also, from case 3 above, it can be seen that the simulation was implemented on the actual INTERTIE system with an almost perfect (equal) division of computational load.

Verification Procedure

Due to the unique nature of the RSRA simulation, it was necessary to perform extensive tests in order to verify that the parallel processing configuration would yield correct results. In order to perform these tests, it was necessary first to develop a single processor version of the RSRA simulation. Based on estimated wind tunnel results, an initial math model was developed by Sikorsky Aircraft. This math model was then programmed into a single processor simulation both at Sikorsky, using machine language on a PDP-11, and at Ames, using FORTRAN on the Xerox Sigma 7.

When the programming was completed, engineers at Sikorsky generated a series of one and two pass subsystem checks on the entire simulated aircraft and provided trim printouts and dynamic strip chart recordings for various flight configurations. These same checks were then performed at Ames and any disparities were cross-checked to determine which simulation was in error. When all such differences were accounted for and corrected, the simulation programs on both computers were assumed to be correct.

TABLE 1.– RSRA TIMING

		Frame time, msec
SIGMA 7 Programs		
AERO, TAIL, TROTOR, ENGINE	(helicopter)	8
	(compound)	14
CONTR7		4
BASIC programs		10
Data transfers		6
DAC's, ADC's, DI/O		4
Scheduler		4
Total	(helicopter)	36
	(compound)	42
SIGMA 8 Programs		
ROTOR, CONTR8	(helicopter, 4bx3s)	24
	(compound, 4bx3s)	28
BASIC programs		4
Data transfers		2
DAC's, ADC's, DI/O		8
Scheduler		4
Total	(helicopter)	42
	(compound)	46

In preparation for the parallel processor checkout at Ames, a final series of trim printouts and dynamic strip chart recordings was generated on the Sigma 7. Examples of results from these printouts and strip chart recordings are contained in appendix E. Finally, all of the subroutines comprising the single processor version were saved on magnetic tape for further generation of any necessary reference data.

The subroutines used in the single-processor checkout were then rearranged to form the INTERTIE loop structure. Once the transfer lists were programmed to provide communication between the two computers, checkout was begun to determine whether the same results were being obtained as for the single processor. After some debugging, correlation between the final static and dynamic checks for the single processor and the parallel processor simulations was obtained.

During the verification procedure certain unusual debugging complications were encountered. The addition of transfer lists leads to numerous potential problems. Race conditions can occur in which parameters of the same name on both computers alternate in value, passing values back and forth through transfer lists. An inadvertant omission of parameters on transfer lists can lead to cases in which a parameter may have one value on one computer and a different value on the other.

Debugging of dynamic instabilities is complicated by the possibility that dynamic instabilities are caused by transfer delays. For example, as was noted earlier, high frequency rotor moments cause an instability in roll acceleration which can only be solved by transferring certain subroutines from the Sigma 7 to the Sigma 8.

For the most part, additional debugging problems caused by the dual computer configuration were not as difficult as had been originally anticipated.

CONCLUDING REMARKS

A real-time, man-in-the-loop, simulation of the Rotor System Research Aircraft was developed and implemented on the Flight Simulator for Advanced Aircraft at Ames Research Center. There were two important results from this effort. First, the simulation provided an additional research tool for the RSRA project that can assist in achieving relatively low cost design development and verification. Secondly, the general capability to perform dual processor simulations has been developed at Ames by combining the Sigma 7 and Sigma 8 computers. This capability to accomplish dual processor simulations alleviates the true-time constraint by computing the program faster. The dual processor approach was 40 percent faster than the single processor implementation. In addition, neither was the time required for daily checkout of the simulation hardware and software increased, nor was the reliability, or net "uptime," of the simulation degraded by use of the dual processor configuration.

Thus, in the situation where a simulation has a critical true-time constraint that cannot be met by a single processor, and when the program is configured from the beginning as a two-computer simulation, the dual processor approach is effective in performing a real-time simulation.

Ames Research Center
National Aeronautics and Space Administration
Moffett Field, California, 94035, July 30, 1976

APPENDIX A

CAB CONFIGURATION

The cab for the RSRA simulation contained two pilot seats plus a seat for an observer or researcher. Both pilot seats had a 48 by 38 cm (19- by 15-in.) television screen for display of the visual scene. Only the pilot seat on the right side had operational controls and instruments.

The following is a list of the pilot controls and instruments:

1. Controls on center console
   - Flap
   - Drag brake
   - Longitudinal Control Phasing Unit
   - Lateral Control Phasing Unit
   - Directional Control Phasing Unit
   - Wing incidence

2. Other controls
   - Collective stick
   - Port propulsion engine rpm (twist grip on collective stick)
   - Starboard propulsion engine rpm (twist grip on collective stick)
   - Longitudinal stick
   - Lateral stick
   - Pedals
   - Main rotor rpm selector

3. Two-way beepers
   - Elevator series trim
   - Aileron series trim
   - Longitudinal cyclic series trim
   - Lateral cyclic series trim
   - Longitudinal CPU series trim
   - Lateral CPU series trim
   - Directional CPU series trim
   - Port propulsion engine series trim
   - Starboard propulsion engine series trim

4. Two-way proportional switches
   - Longitudinal stick parallel trim
   - Lateral stick parallel trim
   - Pedal parallel trim

5. On-off switches or buttons
   - Roll axis SAS
   - Pitch axis SAS

Yaw axis SAS
Main rotor blade release
Event marker
Initial conditions

6. Instruments
   Longitudinal fixed-wing CPU
   Longitudinal rotor CPU
   Lateral fixed-wing CPU
   Lateral rotor CPU
   Directional CPU (double ended needle)
   Flap deflection
   Elevator deflection
   Longitudinal cyclic deflection
   Impressed collective at head
   Rudder deflection
   Port propulsion engine rpm
   Starboard propulsion engine rpm
   Collective stick position
   Pedal position
   Elevator series trim
   Longitudinal cyclic series trim
   Aileron series trim
   Lateral cyclic series trim
   Wing tilt angle
   Main rotor flapping angle
   Blade tip Mach number
   Angle of attack
   Normal acceleration
   Indicated air speed
   Main rotor torque
   Main rotor rpm
   Main rotor port engine rpm
   Main rotor starboard engine rpm
   Sperry attitude and direction indicator
   Directional compass
   Barometric altimeter
   Clock
   Vertical speed
   Side slip
   Turn and bank

In addition to the equipment mentioned above, the RSRA simulation cab contained two speakers connected to a sound generation system to provide wind and engine noise for pilot aural cues.

APPENDIX B

USER COMMON BLOCKS DEVELOPED FOR RSRA

COMMON blocks are utilized in real time simulations on the FSAA in order to facilitate communication among the individual subroutines that are combined to make up the simulation. There are two BASIC COMMON blocks, one for real variables and one for integer variables. For the RSRA, four additional COMMON blocks are defined. Each COMMON block is labeled, and the elements of the block are identified as a numbered element of a one dimensional array. Thus, for example, all of the elements of the COMMON block labeled RSRACOM are identified as elements of a 167 element array named RCM. So, the third element in RSRACOM would be RCM(3). In each subroutine where a given element of a COMMON block is needed, the array element will be equivalenced to the appropriate mnemonic FORTRAN name for easier use.

The four user-defined COMMON blocks for RSRA are:

1. RSRACOM – Elements RCM(1) through RCM(167). This COMMON block is used for communication of real variables internally within the XDS Sigma 7 or the XDS Sigma 8.

2. IRSRA – Elements IRS(1) through IRS(3). This COMMON block is used for communication of integer variables internally within the XDS Sigma 7 or the XDS Sigma 8.

3. ACOUT – Elements ACO(1) through ACO(37). This COMMON block is used to communicate variables from the XDS Sigma 7 (which has the aircraft model) to the XDS Sigma 8 (which has the rotor model).

4. ROTOROUT – Elements RO(1) through RO(20). This COMMON block is used to communicate variables from the XDS Sigma 8 to the XDS Sigma 7.

A complete listing of the above COMMON blocks, with each array element numbered, identified with the appropriate math model quantity and FORTRAN quantity, defined, given units, and point of origin is given in tables 2 through 5.

TABLE 2.– COMMON BLOCK RSRACOM

RCM	Quantity	FORTRAN	Definition	Unit	From
1	$V_{xg} \cdot f(\tau_T)$	DVXG	Body axis wind vel, X, delayed by τ_T	ft/sec	TAIL
2	$V_{yg} \cdot f(\tau_T)$	DVYG	Body axis wind vel, Y, delayed by τ_T	ft/sec	TAIL
3	$V_{zg} \cdot f(\tau_T)$	DVZG	Body axis wind vel, Z, delayed by τ_T	ft/sec	TAIL
4	FSE	FSJT	Fuselage station of jet engines	in.	Data
5	WLE	WLJT	Water line of jet engines	in.	Data
6	BLE	BLJT	Butt line of jet engines	in.	Data
7	V_{xie}	VXIE	Propulsion engine wash, X	ft/sec	TAIL
8	V_{zie}	VZIE	Propulsion engine wash, Z	ft/sec	TAIL
9	$V_{xie} \cdot f(\tau_T)$	DVYIW	Wing wake wash Y, delayed by τ_T	ft/sec	TAIL
10	$V_{zie} \cdot f(\tau_T)$	DVZIW	Wing wake wash Z, delayed by τ_T	ft/sec	TAIL
11	θ_{otr}	THOTR	Impressed tail rotor collective	deg	CONTR7
12	$E_{ktx}(D_{wo}\Omega_t R_t)$	EKXTERM	Rotor downwash term at tail	ft/sec	TAIL
13	$E_{ktz}(D_{wo}\Omega_t R_t)$	EKZTERM	Rotor downwash term at tail	ft/sec	TAIL
14	FSCG	FSCG	Fuselage station of CG	in.	Data
15	WLCG	WLCG	Water line of CG		
16	FSWT	FSWT	Fuselage station of wing		
17	WLWT	WLWT	Water line of wing		
18	FSTR	FSTR	Fuselage station of tail rotor		
19	WLTR	WLTR	Water line of tail rotor		
20	BLTR	BLTR	Butt line of tail rotor		
21	i_w	WINC	Wing incidence	deg	CONTR7
22	X_{wf}	XWF	Body axis x force from wing fuselage, nac.	lb	AERO
23	Y_{wf}	YWF	Body axis y force from wing fuselage, nac.	lb	
24	Z_{wf}	ZWF	Body axis z force from wing fuselage, nac.	lb	
25	L_{wf}	TLWF	Body axis roll moment from wing fuselage, nac.	ft-lb	
26	M_{wf}	TMWF	Body axis pitch moment from wing fuselage, nac.	ft-lb	
27	N_{wf}	TNWF	Body axis yaw moment from wing fuselage, nac.	ft-lb	
28	v_{xgt}	VXGT	Body axis wind & turbulence, X	ft/sec	
29	v_{ygt}	VYGT	Body axis wind & turbulence, Y	ft/sec	
30	v_{zgt}	VZGT	Body axis wind & turbulence, Z	ft/sec	
31	α_{wf}	ALFWF	α of wing – without i_w	deg	AERO
32	β_{wf}	BETAWF	β of wing		AERO
33	i_e	XIJT	Engine shaft angle		Data
34	D_{tot}	TOTD	Drag force total of wing, fus., nac.		AERO
35	Y_{tot}	TOTY	Side force total of wing, fus., nac.		AERO

TABLE 2.– Continued

RCM	Quantity	FORTRAN	Definition	Unit	From
36	L_{mtot}	TML	Roll moment total of wing, fus., nac.	ft/lb	AERO
37	M_{mtot}	TMM	Pitch moment total of wing, fus., nac.	ft/lb	AERO
38	N_{mtot}	TMN	Yaw moment total of wing, fus., nac.	ft/lb	AERO
39	i_{htu}	UHTINC	Upper horizontal tail incidence	deg	BLOCK
40	FSHTU	FSHTU	Fuselage station – upper horizontal tail	ins.	BLOCK
41	WLHTU	WLHTU	Waterline station – upper horizontal tail	ins.	BLOCK
42	K_{qvt}	XKQVT	Dynamic pressure loss factor, vert. tail	ND	TAIL
43	ΔC_{lf}	DCLFF3	Lift increment due to flap deflection	lb	AERO
44	Ψ_{wf}	PSIWF	Wing – fus. yaw angle	deg	AERO
45	T_p+T_s	TJTSUM	Total (port + starboard) propulsion engine thrust	lb	ENGINE
46	FSEI	FSJTI	Fuselage station of propulsion engine inlet	in.	Data
47	θ_{tr}	THETTR	Tail rotor collective pitch	deg	TROTOR
48	δ_f	FLAP	Flap angle		CONTR7
49	δ_a	AIL	Aileron		
50	δ_r	RUD	Rudder		
51	δ_e	ELEV	Elevator		
52	α_w	ALFWG	α at wing + iw		AERO
53	X_{tr}	XTR	Body axis force from tail rotor, X	lb	TROTOR
54	Y_{tr}	YTR	Body axis force from tail rotor, Y	lb	
55	Z_{tr}	ZTR	Body axis force from tail rotor, Z	lb	
56	L_{tr}	TRL	Body axis moment from tail rotor, Roll	ft-lb	
57	M_{tr}	TRM	Body axis moment from tail rotor, Pitch	ft-lb	
58	N_{tr}	TRN	Body axis moment from tail rotor, Yaw	ft-lb	
59	FSHT	FSHT	Fuselage station of lower horizontal tail	in.	Data
60	WLHT	WLHT	Waterline station of lower horizontal tail		
61	FSVT	FSVT	Fuselage station of vertical tail		
62	WLVT	WLVT	Water line station of vertical tail		
63	FSDB	FSDB	Fuselage station of drag brake		
64	WLDB	WLDB	Water line station of drag brake		
65	X_t	XT	Body axis force from empennage, X	lb	TAIL

TABLE 2.– Continued

RCM	Quantity	FORTRAN	Definition	Unit	From
66	Y_t	YT	Body axis force from empennage, Y	lb	TAIL
67	Z_t	ZT	Body axis force from empennage, Z	lb	
68	L_t	TLT	Body axis moment from empennage, Roll	ft-lb	
69	M_t	TMT	Body axis moment from empennage, Pitch	ft-lb	
70	N_t	TNT	Body axis moment from empennage, Yaw	ft-lb	
71	D_{wtr}	DWTR	Downwash from tail rotor	N.D.	TROTOR
72	Ω_{tr}	OMEGTR	Tail rotor angular velocity	rad/sec	TROTOR
73	R_{tr}	RTR	Tail rotor radius	ft	Data
74	i_{ht}	XIHT	Lower horizontal tail incidence	deg	TAIL
75	ALFWFR	ALFWFR	Angle of attack – body axes	rad	AERO
76	q_{wf}	QWF	Dynamic pressure at wing-fuselage	lb/ft^2	AERO
77	δ_{db}	DRAG	Drag brake angle	deg	CONTR7
78	X_E	XJT	Body axis force from propulsion engines, X	lb	ENGINE
79	Y_E	YJT	Body axis force from propulsion engines, Y	lb	
80	Z_E	ZJT	Body axis force from propulsion engines, Z	lb	
81	L_E	TLJT	Body axis moment from propulsion engines, Roll	ft-lb	
82	M_E	TMJT	Body axis moment from propulsion engines, Pitch	ft-lb	
83	N_E	TNJT	Body axis moment from propulsion engines, Yaw	ft-lb	
84	V_{xwf}	VXWF	Wing-fuselage velocity component, X	ft/sec	AERO
85	V_{ywf}	VYWF	Wing-fuselage velocity component, Y	ft/sec	AERO
86	V_{zwf}	VZWF	Wing-fuselage velocity component, Z	ft/sec	AERO
87	XATRM	XATRM	Trim value of lateral stick	percent	Trim or data
88	XBTRM	XBTRM	Trim value of longitudinal stick		
89	XCTRM	XCTRM	Trim value of collective		
90	XPTRM	XPTRM	Trim value of pedals		
91	CDELA	CDELA	CPU gain to ailerons	deg/deg	CONTR7
92	CDELE	CDELE	CPU gain to elevators	deg/deg	CONTR7
93	CTTR	CTTR	CPU gain to tail rotor	deg/deg	CONTR7

TABLE 2.– Continued

RCM	Quantity	FORTRAN	Definition	Unit	From
94	CA1S	CA1S	CPU gain to lateral rotor control	deg/deg	CONTR8
95	CB1S	CB1S	CPU gain to longitudinal rotor control	deg/deg	CONTR8
96	PRPM(left)	RPMJETP	Actual jet engine RPM (port engine)	percent	ENGINE
97	TKQB	TKB	Rotor longitudinal SAS lagged pitch rate time constant	sec	Data
98	RKQB	RKB	Rotor longitudinal SAS pitch rate gain	deg/deg/deg	Data
99	LRKQB	BLRK	Rotor longitudinal SAS lagged pitch rate gain	deg/deg/deg	
100	TKQE	TKE	Elevator SAS lagged pitch rate time constant	sec	
101	RKQE	RKE	Elevator SAS pitch rate gain	deg/deg/deg	
102	LRKQE	ELRK	Elevator SAS lagged pitch rate gain	deg/deg/deg	
103	TKPS	TKA1	Rotor lateral SAS lagged roll rate time	sec	
104	RKPS	RKA1	Rotor lateral SAS lagged roll constant	sec	
104	RKPS	RKA1	Rotor lateral SAS roll rate gain	deg/deg/sec	
105	LRKPS	A1LRK	Rotor lateral SAS lagged roll rate gain	deg/deg/sec	
106	TKPL	TKA	Aileron SAS lagged roll rate time constant	sec	
107	RKPL	RKA	Aileron SAS roll rate gain	deg/deg/sec	
108	LRKPL	ALRK	Aileron SAS lagged roll rate gain	deg/deg/sec	
109	TKRT	TK5T	Tail rorot SAS lagged yaw rate time constant	sec	
110	RKRT	RK8T	Tail rotor SAS yaw rate gain	deg/deg/sec	
111	LRKRT	TLRK5	Tail rotor SAS lagged yaw rate gain	deg/deg/sec	
112	TKRR	TK5R	Rudder SAS lagged yaw rate time constant	sec	
113	RKRR	RK8R	Rudder SAS yaw rate gain	deg/deg/sec	
114	LRKRR	RLRK5	Rudder SAS lagged yaw rate gain	↓	↓
115	RKRPS	RKYA1	Rotor lateral SAS yaw rate gain		
116	RKRPL	RKYA	Aileron SAS yaw rate gain		

TABLE 2.– Continued

RCM	Quantity	FORTRAN	Definition	Unit	From
117	RKPRT	RKR7T	Tail rotor SAS roll rate gain	deg/deg/sec	Data
118	RKPRR	RKR7R	Rudder SAS roll rate gain	deg/deg/sec	
119	W_{ob}	WOB	Rotor longitudinal SAS wash-out constant	Hz	
120	W_{oe}	WOE	Elevator SAS wash-out constant	Hz	
121	$\sin(\alpha_{wf})$	SALFW	Sine of wing-fuselage angle of attack	N.D.	AERO
122	$\cos(\alpha_{wf})$	CALFW	Cosine of wing-fuselage angle of attack	N.D.	AERO
123	X_p	DPEDAL	Pedal position	percent	CONTR7
124	δ_{pe}(left)	RPMP	Pilot commanded RPM (port engine)		CONTR7
125	XRPMJT(left)	XRPMJTP	Pilot RPM stick position (port engine)		Pilot
126	δ_{pe}(right)	RPMS	Pilot commanded RPM (starboard engine)		CONTR7
127	XRPMJT(right)	XRPMJTS	Pilot RPM stick position (starboard)		Pilot
128	PRPM(right)	RPMJETS	Actual jet engine RPM (starboard)		ENGINE
129	D/q(tail off)	CDF3	Wing-fuselage drag due to angle of attack	lb	MAPIII
130	L/q(tail off)	CLF3	Wing-fuselage lift due to angle of attack		MAPIII
131	Weight	WEIGHT	Total weight of A/C (including blades)		BLOCK
132	T_p	TJTP	Net thrust – port engine		ENGINE
133	T_s	TJTS	Net thrust – starboard engine		ENGINE
134	X_c	COLSTK	Collective stick position	percent	Pilot
135	KCPULG	GKCPULG	Longitudinal CPU lever gearing	N.D.	Data
136	KCPULT	GKCPULT	Lateral CPU lever gearing	N.D.	Data
137	A_{1S}	A1S	Total lateral cyclic at the rotor head	deg	CONTR8
138	B_{1S}	B1S	Total longitudinal cyclic at the rotor head		CONTR8
139	θ_{cuff}	THETAO	Collective pitch cyclic at the rotor head		CONTR7 CONTR8
140	THOL	THOL	Lower limit on collective pitch		Data
141	THOU	THOU	Upper limit on collective pitch		Data
142	ΣBCPULG·ΔT	CPLGTRM	Longitudinal CPU trim (from beeper)	percent	CONTR8
143	ΣBCPULT·ΔT	CPLTTRM	Lateral CPU trim (from beeper)	percent	CONTR8
144	CPUDIR	CPUDIR	Directional CPU position	percent	CONTR7

TABLE 2.– Concluded

RCM	Quantity	FORTRAN	Definition	Unit	From
145	ELEV1	ELVTRM	Elevator series trim	deg	CONTR7
146	AIL1	AILTRM	Aileron series trim	deg	CONTR7
147	L_{wt}	TØTLWT	Total lift, wind tunnel axes	lb	FORCE
148	D_{wt}	TØTDWT	Total drag, wind tunnel axes	lb	FORCE
149	KASAIL	XKASAIL	Aileron asymmetric gearing ratio	N.D.	Data
150	XAILG	XAILG	Aileron asymmetric gearing ratio gain	N.D.	Data
151	f(a) (long)	GGRADLO	Longitudinal stick gradient	lb/in.	CONTR7
152	f(b) (long)	DDAMPLO	Longitudinal stick damping	lb/in./sec	
153	f(c) (long)	ACRTLO	Longitudinal stick force dependent on A/C speed	lb/deg/sec	
154	f(a) (lat.)	GGRADLA	Lateral stick gradient	lb/in.	
155	f(b) (lat.)	DDAMPLA	Lateral stick damping	lb/in./sec	
156	f(c) (lat.)	ACRTLA	Lateral stick force dependent on A/C speed	lb/deg/sec	
157	f(a) (dir.)	GRADIN	Pedal gradient	lb/in.	
158	f(b) (dir.)	DAMPDIN	Pedal damping	lb/in./sec	
159	K_{EG}	GKEG	Elevator – lower horizontal tail gain	deg/deg	
160	CTW	CTW	Wing – lower horizontal tail nonlinear constant	deg	
161	KTWI	XKTWI	Wing – lower horizontal tail nonlinear gain	N.D.	Data
162	KTW2	XKTW2	Wing – lower horizontal tail nonlinear gain	deg^{-1}	
163	KTW5	XKTW5	Wing – lower horizontal tail nonlinear gain	deg^{-4}	
164	KTLIN1	XKTLIN1	Lower horizontal tail linearization gearing coefficient	N.D.	
165	KTLIN5	XKTLIN5	Lower horizontal tail linearization gearing coefficient	deg^{-4}	
166	CTFLAP	CTFLAP	Flap gearing coefficient	deg	
167	KTFLAP	XKTFLAP	Flap gearing coefficient	N.D.	

TABLE 3.– COMMON BLOCK IRSRA

IRSRA	Quantity	FORTRAN	Definition	Unit	From
1	ICONFIG	ICONFIG	Configuration switch	N.D.	Data or engineer
2	LAGLIM	LAGLIM	Limit rotor lag to ±15° for 60 passes		Trim
3	NOROT	NOROT	No main rotor		ROTOR
4	NOTROT	NOTROT	No tail rotor		TAIL
5	ITRLIM	ITRLIM	Trim value of a control has reached a limit		Trim
6	NCYC1	NCYC1	Σ7 cycle indicator		UTIL7
7	NCYC2	NCYC2			
8	NCYC3	NCYC3			
9	NCYC4	NCYC4			
10	A	IRLSWA	Fix long. and lat. CPU's at 33 percent following rotor release		Data or engineer
11	B	IRLSWB	Fix tail rotor and rudder mixing gains to 0 following rotor release		
12	NOROTIC	NOROTIC	Fixed wing configuration – no rotor (from I.C.)		
13	ISEHO	ISEHO	Elevator SAS hardover		
14	ISAHO	ISAHO	Aileron SAS hardover		
15	ISRHO	ISRHO	Rudder SAS hardover		
16	ISTRHO	ISTROHO	Tail rotor SAS hardover		
17	ISBHO	ISBHO	Longitudinal cyclic SAS hardover		
18	ISA1HO	ISA1HO	Lateral cyclic SAS hardover		
19	ITRHO	ITRHO	Tail rotor hardover		
20	IFLPHO	IFLPHO	Flap hardover		
21	IWNGJM	IWNGJM	Wing incidence jam		
22	IWNGHO	IWNGHO	Wing incidence hardover		
23	IDRGF	IDRGF	Drag brake failure (±15° from present position)		
24	IFLO	IFLO	Longitudinal stick force feel system failure		
25	IFLA	IFLA	Lateral stick force feel system failure		
26	IFDR	IFDR	Pedal force feel system failure		
27	IJTPFL	IJTPFL	Port engine failure		
28	IJTSFL	IJTSFL	Starboard engine failure		
29	NOTROTIC	NOTROTIC	No tail rotor (from I.C.)		
30	NOTROTSW	NOTROTSW	Tail rotor severance failure switch		

TABLE 4.– COMMON BLOCK ACOUT

ACOUT	Quantity	FORTRAN	Definition	Unit	From
1	V_{xb}	VXB	Body axis vel. of CG without wind, X	f/sec	AERO
2	V_{yb}	VYB	Body axis vel. of CG without wing, Y		
3	V_{zb}	VZB	Body axis vel. of CG without wind, Z		
4	$\mathring{V}_{xb}$	VXBDOT	Body axis acceleration of CG without wind, X		
5	$\mathring{V}_{yb}$	VYBDOT	Body axis acceleration of CG without wind, Y		
6	$\mathring{V}_{zb}$	VZBDOT	Body axis acceleration of CG without wind, Z		
7	V_{xg}	VXG	Body axis vel. of wind, X		
8	V_{yg}	VYG	Body axis vel. of wind, Y		
9	V_{zg}	VZG	Body axis vel. of wind, Z		
10	—	AC1	—	—	—
11	CPULGFX	CPULGFX	Long. CPU drive for cab instrument	ft/sec	CONTR7
12	CPULTFX	CPULTFX	Lat. CPU drive for cab instrument	ft/sec	CONTR7
13	δ_{re}	DRE	Rotor speed control	N.D.	CONTR7
14	L_{tot}	TOTL	Lift in wing axis of wing-fus-nac.	lb	AERO
15	GRADLO	GRADLO	Total longitudinal stick gradient	lb/in.	CONTR7
16	GRADLA	GRADLA	Total lateral stick gradient	lb/in.	
17	GRADI	GRADI	Total pedal gradient	lb/in.	
18	DAMPLO	DAMPLO	Total longitudinal stick damping	lb/in./sec	
19	DAMPLA	DAMPLA	Total lateral stick damping	lb/in./sec	
20	DAMPDI	DAMPDI	Total pedal damping	lb/in./sec	
21	BIASLO	BIASLO	Longitudinal stick bias	lb	
22	BIASLA	BIASLA	Lateral stick bias		
23	BIASDR	BIASDR	Pedal bias		
24	BOLO	BOLO	Longitudinal stick break-out		
25	BOLA	BOLA	Lateral stick break-out		
26	BODI	BODI	Pedal break-out		
27	HSTLO	HSTLO	Longitudinal stick hysteresis		
28	HSTLA	HSTLA	Lateral stick hysteresis		
29	HSTDI	HSTDI	Pedal hysteresis		
30	TLOIN	TLOIN	Longitudinal stick parallel trim position – inches	in.	
31	TLAIN	TLAIN	Lateral stick parallel trim position – inches	in.	
32	TDIIN	TDIIN	Pedal parallel trim position – inches	in.	

TABLE 4.– Concluded

ACOUT	Quantity	FORTRAN	Definition	Unit	From
33	STOPLO	STOPLO	Computed stop – longitudinal stick	in.	CONTR7
34	STOPLA	STOPLA	Computed stop – lateral stick	in.	CONTR7
35	STOPDR	STOPDR	Computed stop – pedals	in.	CONTR7
36	α_{wf}-$\alpha_{wf_{trim}}$	ALFDIF	Difference between actual α and trimmed α	deg	AERO
37	β_{wf}-$\beta_{wf_{trim}}$	BETDIF	Difference between actual β and trimmed β	deg	AERO

TABLE 5.– COMMON BLOCK ROTOROUT

RO COMMON	Quantity	FORTRAN	Definition	Unit	From
1	D_{wo}	DOWNW	Uniform component of rotor downwash	N.D.	ROTOR
2	Ω_t	ØMEGAM	Rotor angular velocity (trim)	rad/sec	
3	R_t	RMR	Rotor radius	ft	
4	Ω	ØMGMR	Actual rotor angular velocity	rad/sec	
5	X_{mr}	XMR	Rotor body axis force, X	lb	
6	Y_{mr}	YMR	Rotor body axis force, Y	lb	
7	Z_{mr}	ZMR	Rotor body axis force, Z	lb	
8	L_{mr}	RML	Rotor body axis moment, L	ft-lb	
9	M_{mr}	RMM	Rotor body axis moment, M	ft-lb	
10	N_{mr}	RMN	Rotor body axis moment, N	ft-lb	
11	χ	CHI	Rotor wake skew angle	deg	
12	λ	XLAMDA	Rotor inflow	N.D.	
13	NBS	NBS	Number of blades simulated	N.D.	
14	NSS	NSS	Number of segments simulated	N.D.	
15	β	BR	Blade flapping angle	rad	
16	δ	XLAG	Blade lagging angle	rad	
17	XB1SEQ	XB1SEQ	Longitudinal stick equiv. position due to B1S series trim	percent	CONTR8
18	XA1SEQ	XA1SEQ	Lateral stick equiv. position due to A1S series trim	percent	CONTR8
19	Ω/Ω_t	OMGRAT	Ratio of actual to trimmed rotor speed	N.D.	ROTOR
20	$\bar{Q}$	QBARMR	Filtered rotor moment – yaw	ft-lb	ROTOR

APPENDIX C

USER-SUPPLIED SUBROUTINES

AERO

The AERO subroutine (fig. 7) obtains body velocities and accelerations and wind velocities in local frame axes (ref. 3) as calculated by BASIC subroutines. These velocities and accelerations are translated to body axes. Wing and wing-fuselage angle of attack and sideslip are computed. Based on configuration, a branch is made to the appropriate aerodynamic map tables and calculations. Upon return, the effects of the stability derivatives are computed and then wind axis forces and moments are computed and rotated to body axes.

BATMOSPH

BATMOSPH (fig. 8) computes air density and speed of sound as a linear function of altitude, and computes Mach number, dynamic pressure, and equivalent air speed in knots.

CONTR7

The CONTR7 subroutine (fig. 9) computes all control surface deflections, propulsion engine rpm command and rotor speed ratio, but does not include calculations of main rotor longitudinal and lateral cyclic command. Stability augmentation system (SAS) inputs are calculated for elevator, aileron, rudder, and tail rotor. Pilot beeper trim inputs are computed for elevator, aileron, rotor shaft, TF-34 fan speed, and directional Control Phasing Unit (CPU). Pilot beeper trim, and inputs for longitudinal and lateral CPUs are obtained from CONTR8 for computation of elevator and aileron deflection, respectively. The CPU null system, which attempts to prevent control surface movement when the pilot is changing CPU gain, is computed for elevator and aileron deflection. Pilot CPU gains are computed for elevator, aileron, rudder and tail rotor. Force feel system parameters are also computed to drive the programmable control force loader. Gradient, trim position, damping, pilot bias, breakout, stops, and hysteresis are calculated for longitudinal and lateral stick, and for pedals.

CONTR8

The CONTR8 subroutine (fig. 10) computes main rotor collective, and longitudinal and lateral cyclic commands for use in conjunction with the ROTOR subroutine on the XDS Sigma 8 computer. Pilot beeper inputs for lateral and longitudinal cyclic series trim and CPU gain are computed. CPU null system and CPU gains are calculated for longitudinal and lateral cyclic.

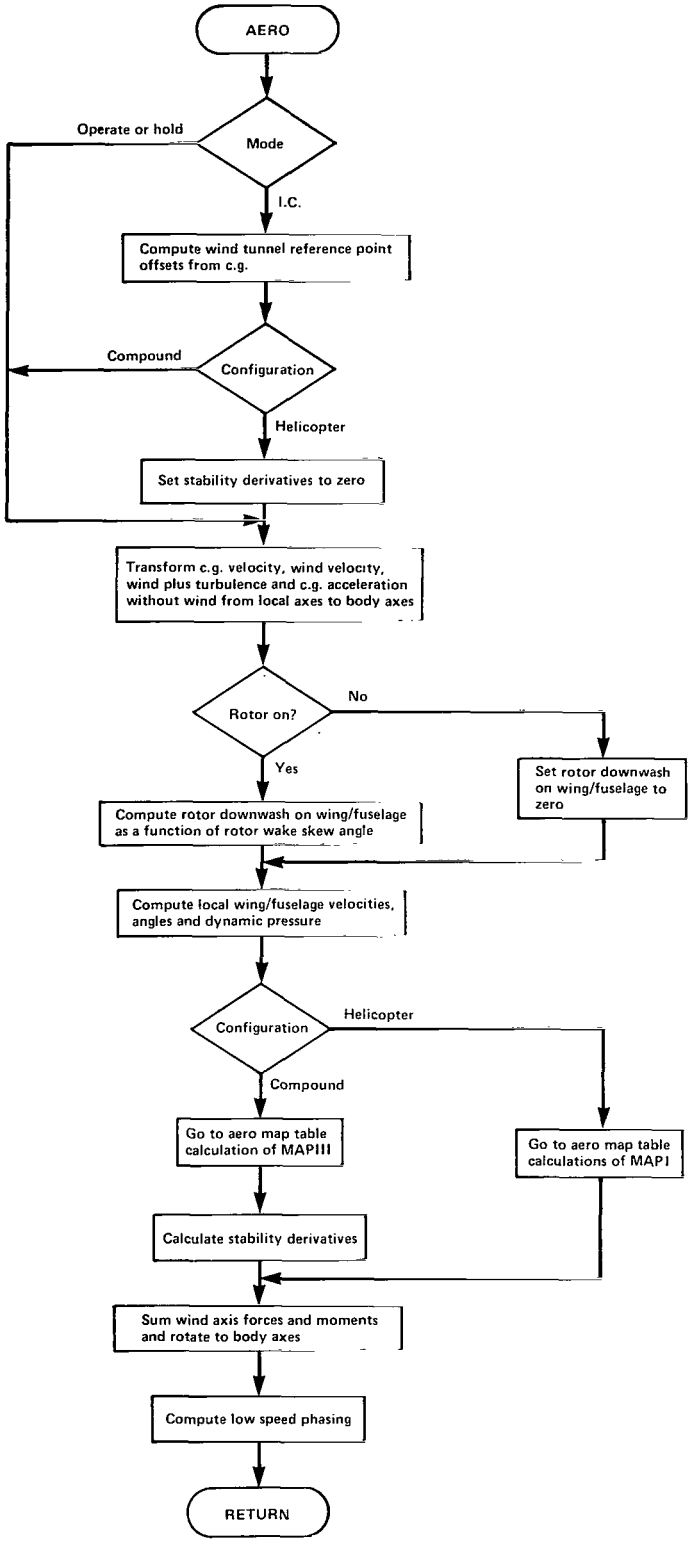


Figure 7.– AERO subroutine.

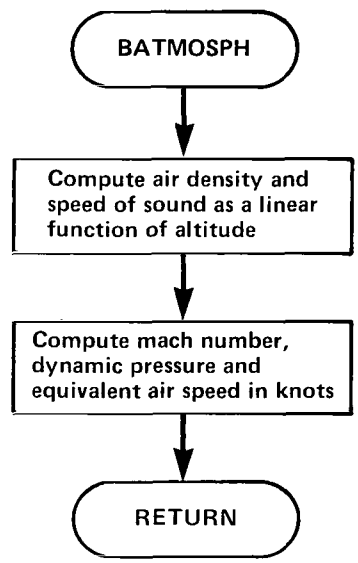


Figure 8.– BATMOSPH subroutine.

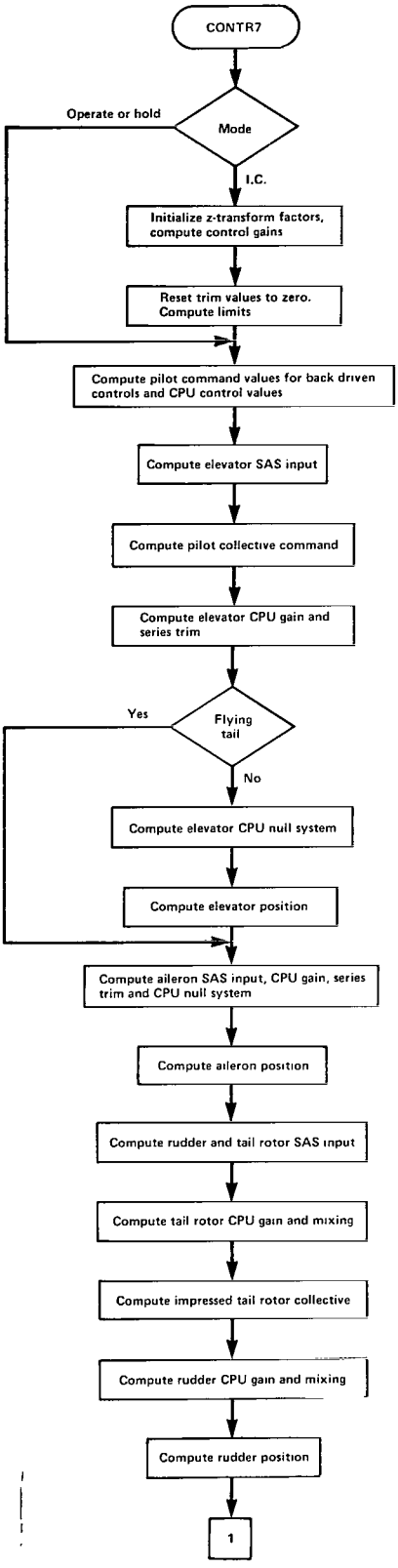


Figure 9.– CONTR7 subroutine.

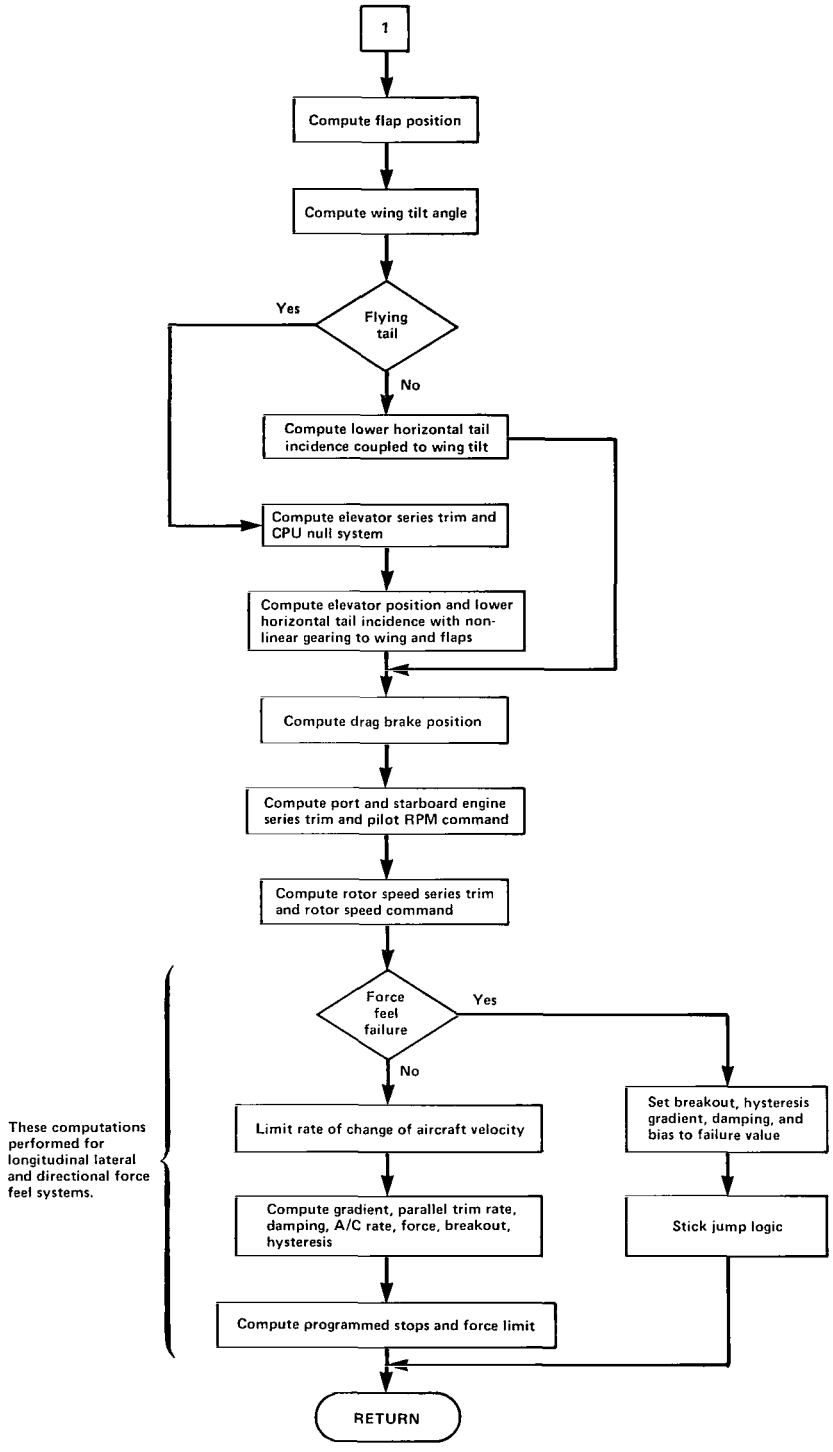


Figure 9.– Concluded.

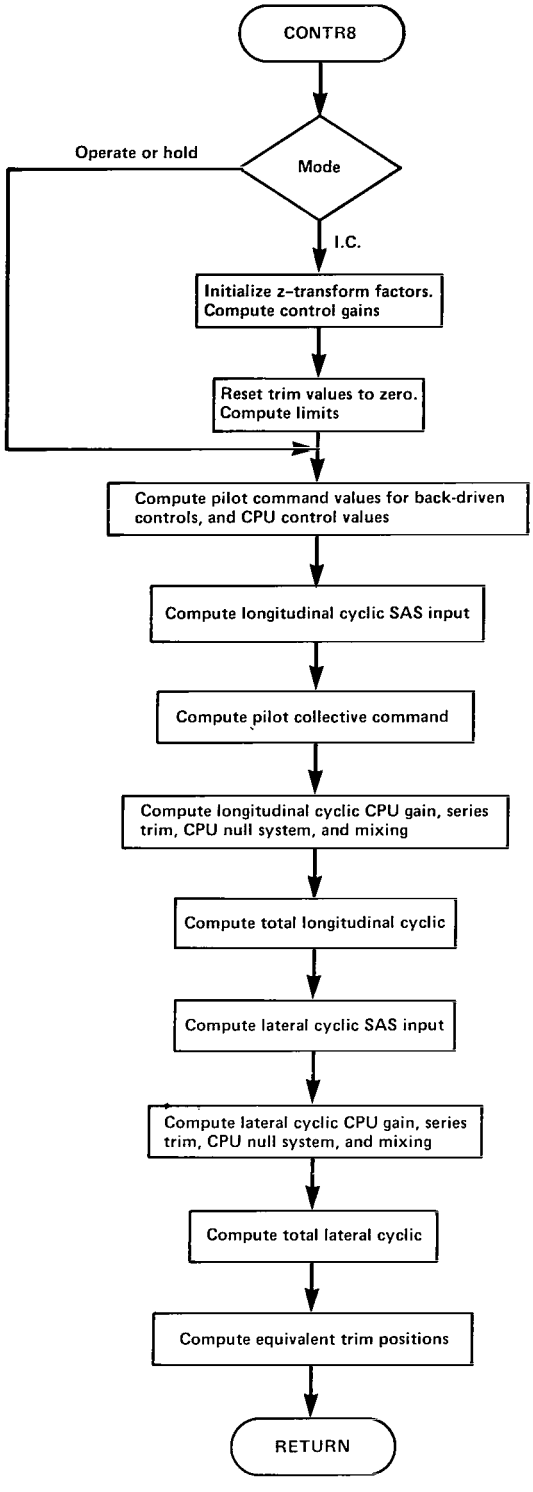


Figure 10.– CONTR8 subroutine.

Equivalent lateral and longitudinal cyclic series trim positions are computed for use in calculation of control stick deflection limits in CONTR7.

ENGINE

The engine model (fig. 11) is designed to match static and dynamic performance data of the TF-34 engine. Computations for port and starboard propulsion engine rpm, windmill drag, thrust, and mass flow are identical. Increases in pilot-commanded rpm result in a pure time delay dependent on commanded rpm and existing rpm, and in a first order lag dependent on delayed rpm and existing rpm. Decreases in pilot-commanded rpm result only in a first order lag. The pure time delay with increases in pilot commanded rpm is implemented using an array of 100 elements and a standard time delay algorithm. Decreases in pilot rpm require that the entire delay array be updated. Because of time requirements to accomplish this update of all 100 elements, a special set of pointers has been developed which gives the same results but only requires the update of one table element and simple logic statements. Engine failure characteristics for each engine are accounted for individually, permitting failure of either or both propulsion engines. Thrust, mass flow, and windmill drag values are combined with geometry terms to compute propulsion engine forces and moments.

FORCE

The FORCE subroutine (fig. 12), which resides on the XDS Sigma 7, sums forces and moments, except rotor moments, in body axes. These forces and moments are also rotated to wind tunnel axes for data output.

INS10AB

The INS10AB subroutine (fig. 13) computes drive signals for the following instruments:

1. Altitude rate
2. Glide slope and localizer
3. Heading
4. Heading discrete.

MAPI

The MAPI subroutine (fig. 14) is called from AERO in helicopter mode. From wind tunnel map data and equations, the fuselage lift, drag, side force, rolling moment, pitching moment, and yawing moments are computed based on angle of attack and sideslip.

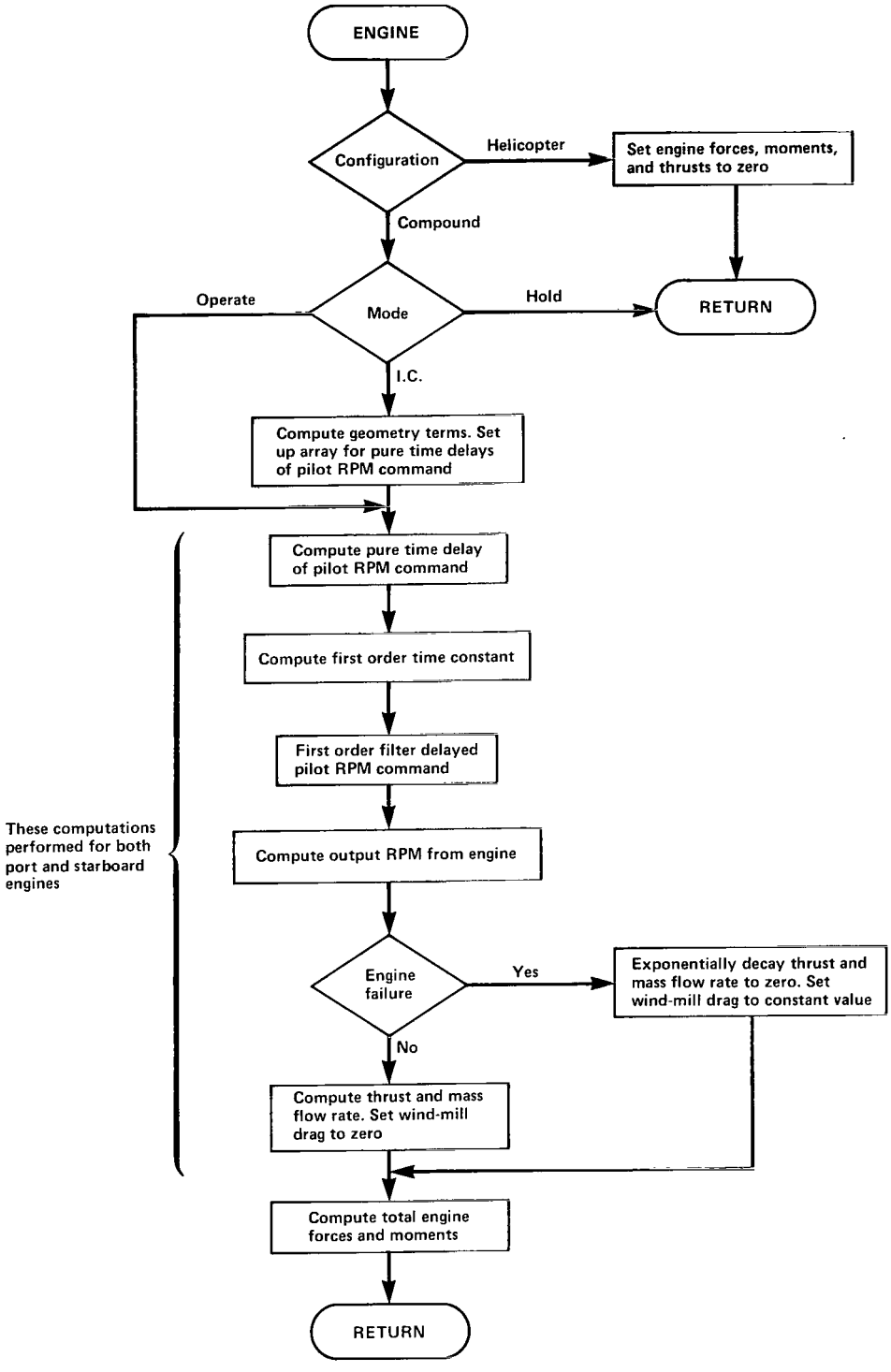


Figure 11.– Engine subroutine.

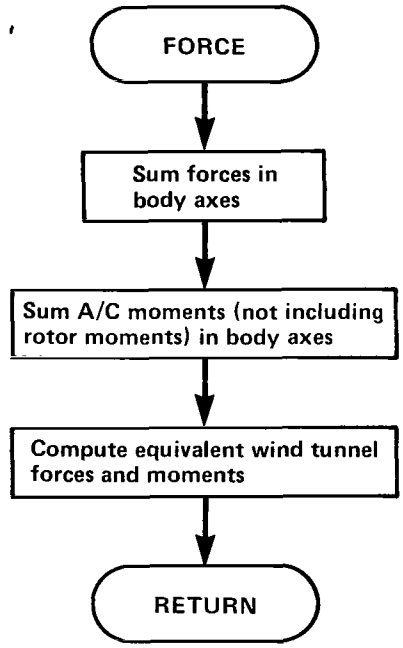


Figure 12.– FORCE subroutine.

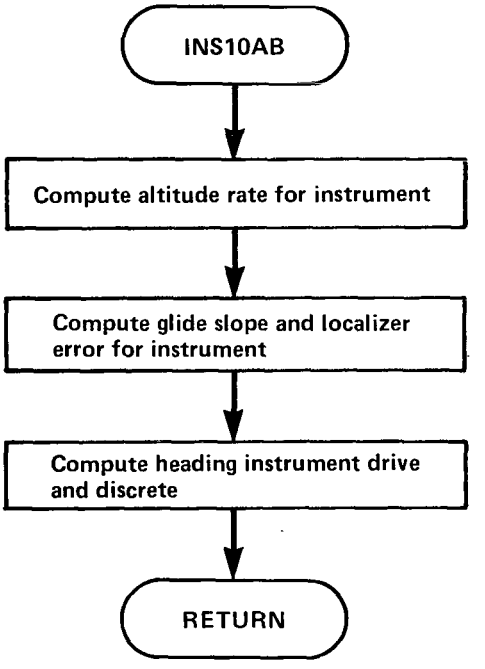


Figure 13.– INS10AB subroutine.

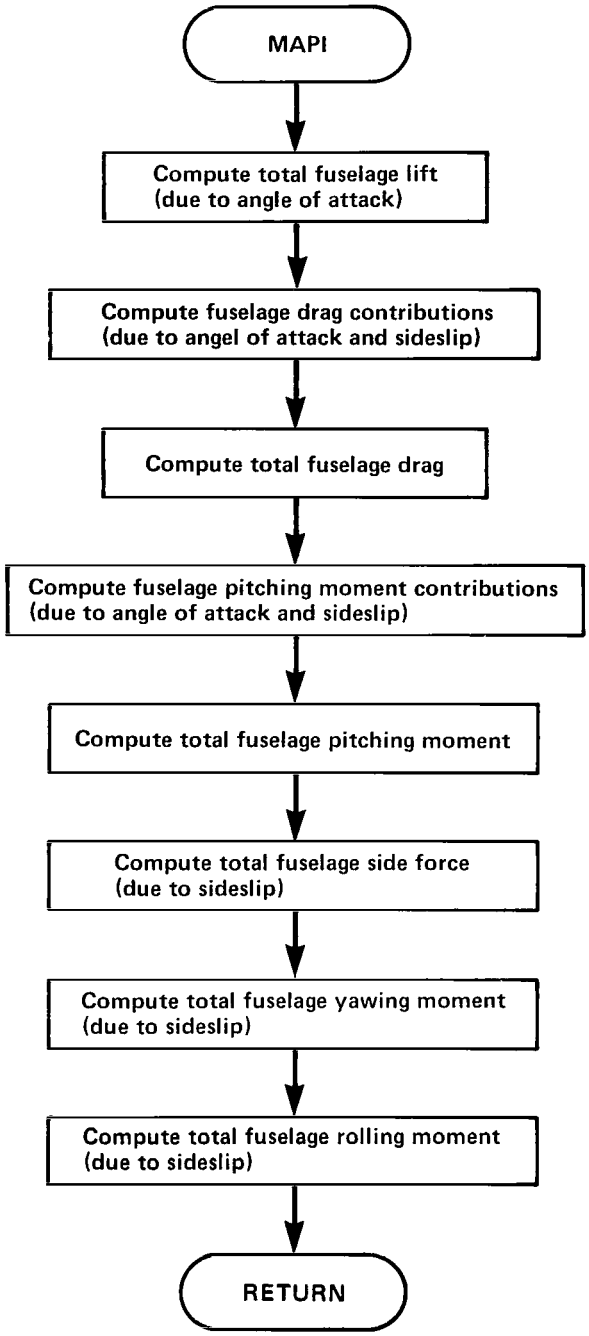


Figure 14.– MAPI subroutine.

MAPIII

The MAPIII subroutine (fig. 15) is called from AERO in fixed-wing or compound mode. From wind tunnel data maps and equations, the fuselage lift, drag, side force, rolling moment, pitching moment, and yawing moment are calculated. Computation of these forces and moments takes into account contributions due to angle of attack, sideslip, propulsion engines, aileron deflection and flap deflection.

ROTOR

The ROTOR subroutine (fig. 16) is the major subroutine on the XDS Sigma 8, and is, in fact, the most time-consuming subroutine in the simulation. This subroutine determines main rotor forces and moments based on aerodynamic and inertia loads acting on each simulated blade. The number of simulated blades may have to be less than the five blades in the actual RSRA due to real-time blade azimuth advance considerations. Each simulated blade is divided into segments which sweep out equal elemental areas. This procedure is used in the real-time program in order to allow minimization of the number of segments, reduction of input data, and distribution of segments toward areas of higher dynamic pressure. The rotor representation allows for flapping and lagging degrees of freedom, and, when the rotor speed governor is released, for rotor shaft degree of freedom. Aerodynamic loads are determined using blade element theory. During one pass through the subroutine, each simulated blade is considered independently, with forces computed for each segment along the blade. If the number of blades simulated is not equal to the actual number of blades in the main rotor, the results of the computations are accordingly distributed in azimuth. With each new pass through the subroutine, the azimuth position of each blade is updated by a factor equal to the simulation's frame time multiplied by rotor speed. Thus, for example, with a rotor speed (Ω) of 21.29 rad/sec and a frame time (Δt) of 30 msec, blade azimuth update angle ($\Delta\Psi = \Omega \cdot \Delta t$) would be $36.6°$.

Total shear forces for each blade are obtained as a result of aerodynamic and inertia shears at the hinge. These forces are summed for all blades and from the results, rotor forces and moments are obtained, which may be filtered to obtain final rotor forces and moments. The subroutine allows for rotor power failure, blade damage, blade failure, and blade release. In addition to total forces and moments, outputs include flapping and lagging angles, velocities, and accelerations, downwash, inflow, rotor wake skew angle, and Fourier coefficients for flapping and lagging. For a more detailed description of the rotor math model, see the RSRA simulation model, vol. I, pp. 5,6,8–13.[2]

SETUP7

The SETUP7 subroutine (fig. 17) is the initialization program for the Sigma 7 programs and is executed in I.C. mode only. It contains logic for the call to GENTRM6, a six-axis trim routine. It computes scales and biases for DAC's and ADC's. The initialization calls for data transfers are made with logic for one time execution. There is provision for sweeping any variable between two limits for data table checks. Finally, a call is made to BSETUP, the BASIC initialization routine.

---

[2]See footnote 1, page 1.

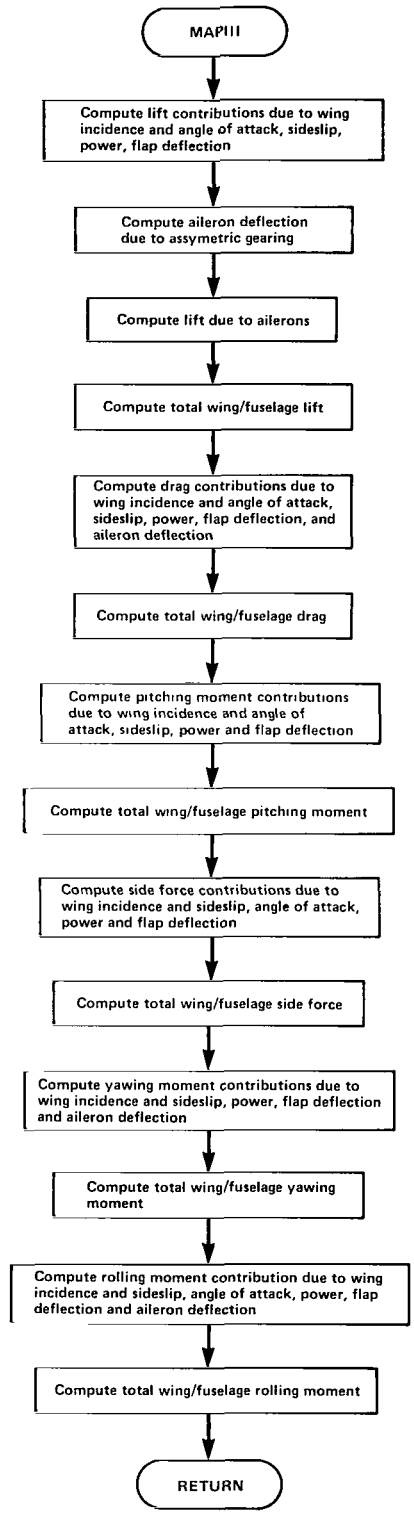


Figure 15.– MAPIII subroutine.

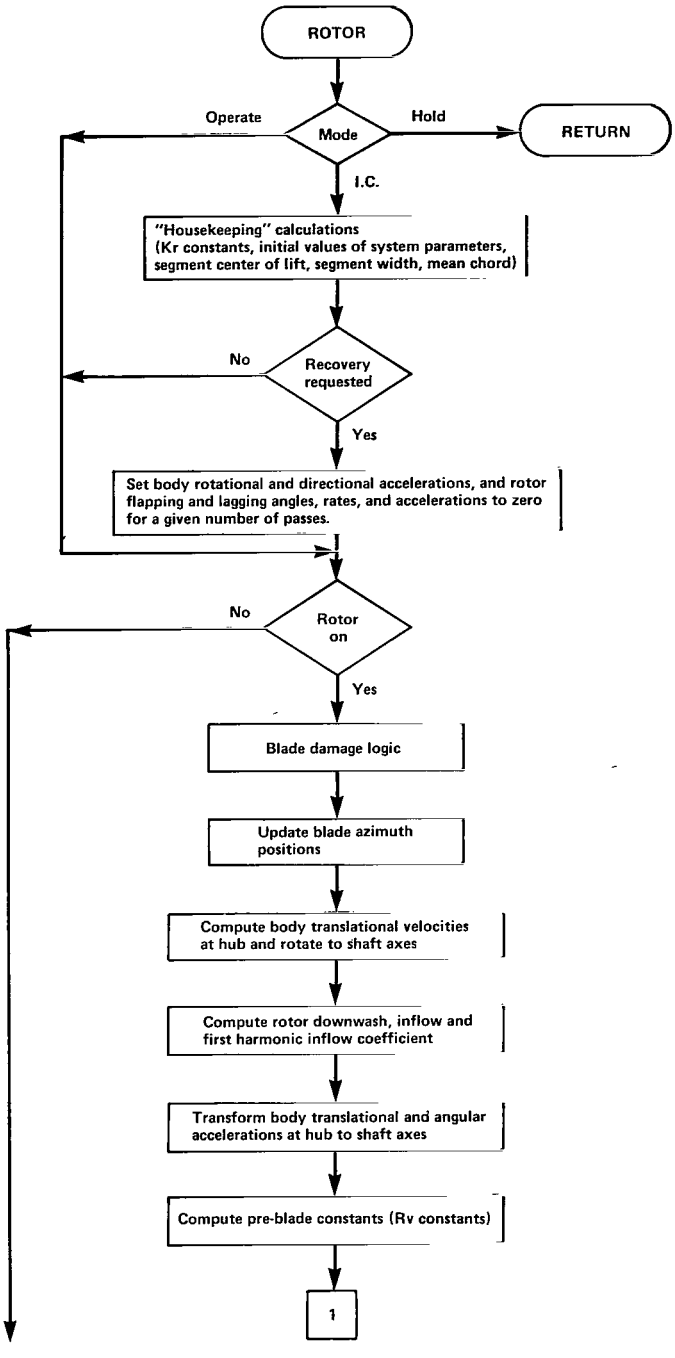


Figure 16.– ROTOR subroutine.

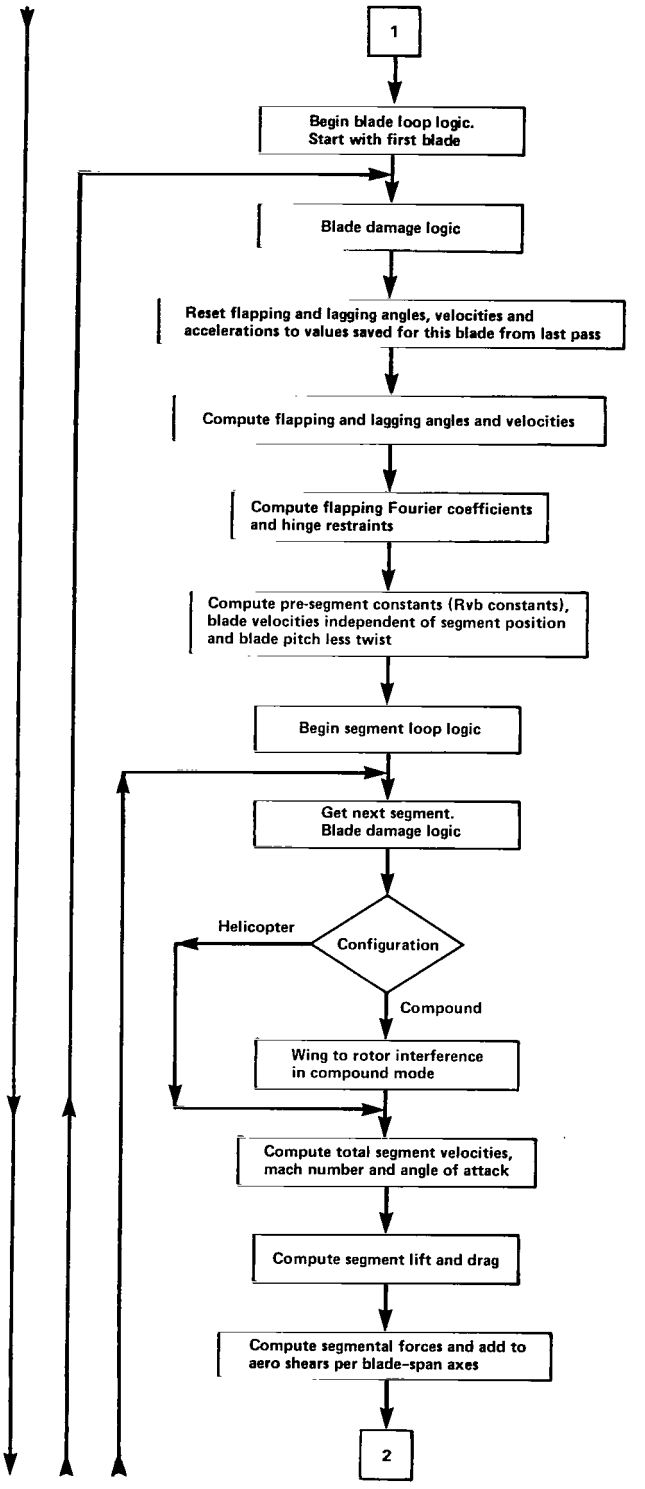


Figure 16.– Continued.

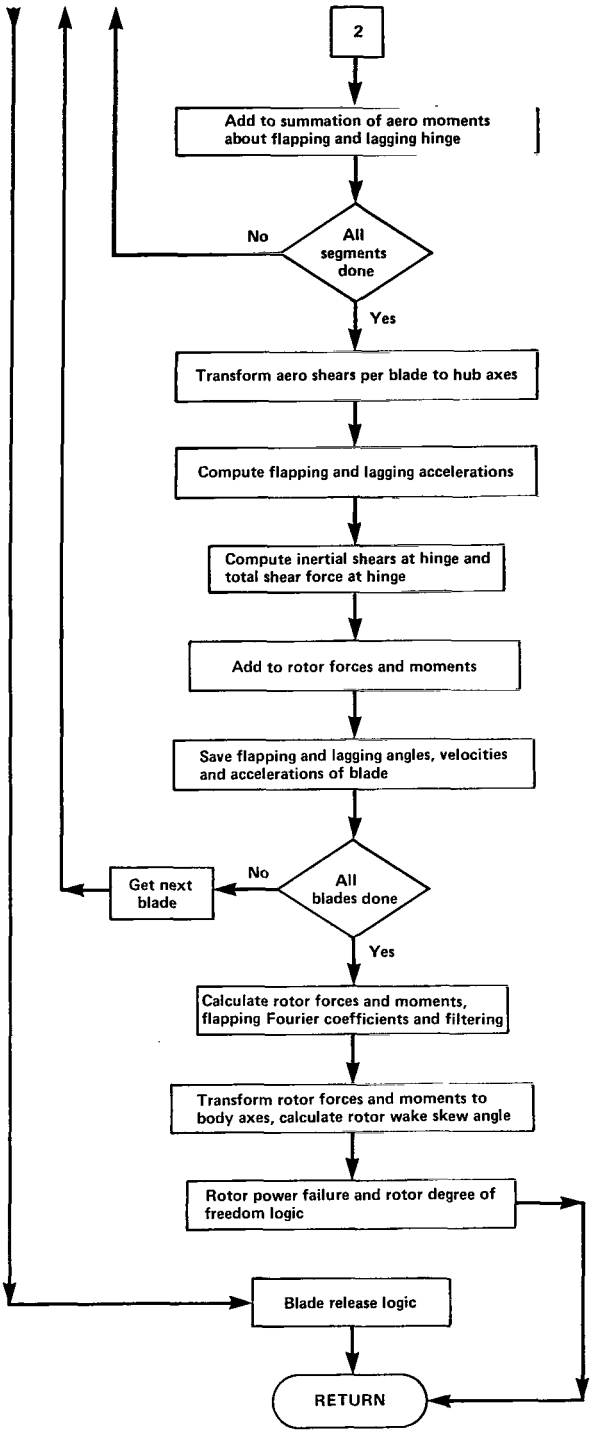


Figure 16.– Concluded.

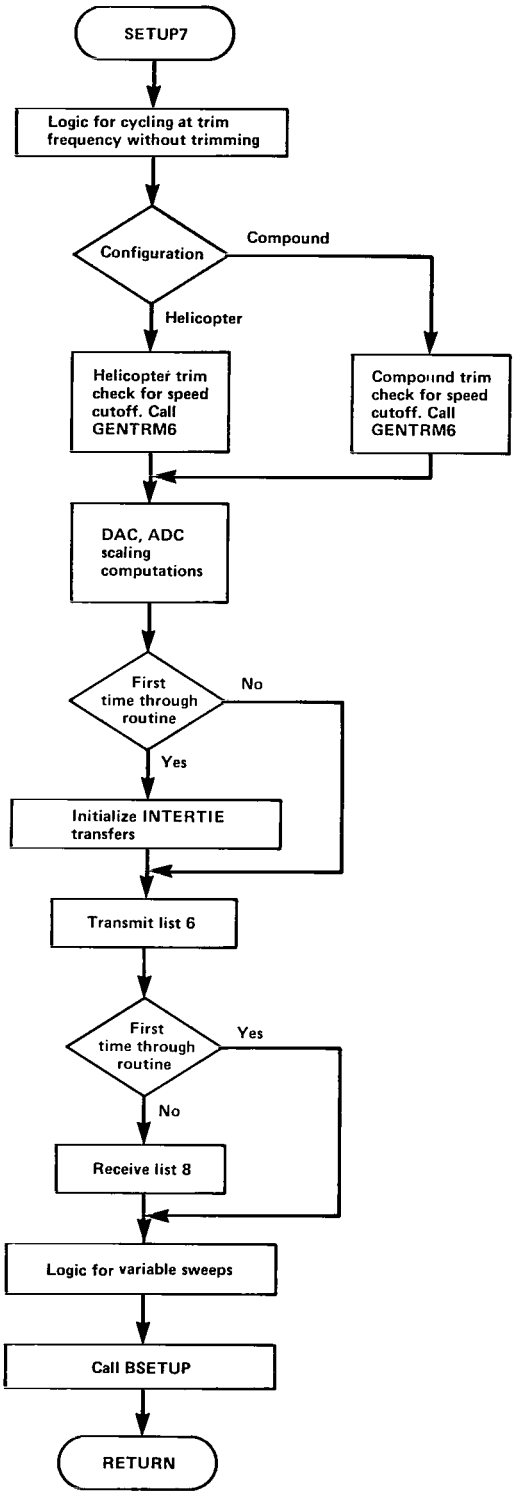


Figure 17.– SETUP7 subroutine.

SETUP8

The SETUP8 subroutine (fig. 18) is the initialization program for the subroutines on the Sigma 8. The transfer lists that go between the computers are initialized on the first pass through the program. There is also some first-pass logic for the I.C. transfer lists to prevent transfer of an uninitialized list. The Euler angles are initialized and pilot station coordinates are computed. Pilot station coordinates must be computed for motion and visual system drives, and must be updated whenever configuration (helicopter or compound) changes occur, because the aircraft center of gravity changes position depending on configuration. Finally, DAC and ADC scales and biases are computed. It should be noted that this program is only executed in I.C.

TAIL

The empennage model (fig. 19) has been separated from the main airframe in order to facilitate tail configuration changes. This flexibility is necessary due to the fact that, during the simulation exercise, tail configuration is frequently altered.

With rotor attached, rotor induced velocities are calculated based on rotor downwash, angular rate, radius, and downwash. Empennage downwash and sidewash velocities are computed. Dynamic pressure loss, including airframe and propulsion engine thrust effects, is calculated, and delayed wind velocities are computed. Based on these values, plus main airframe directional and rotational velocities, local velocities, pressures, angles of attack, and finally local forces are calculated for upper and lower horizontal tails, vertical tail, and drag brake. These forces are transformed into body axes and combined to obtain total tail forces and moments (fig. 18).

TORQ8

The TORQ8 subroutine (fig. 20), which resides on the XDS Sigma 8, obtains aircraft moments from FORCE, and main rotor moments from ROTOR. These moments are summed to produce total vehicle moments.

TROTOR

The TROTOR subroutine (fig. 21) computes force and moments contributions to the vehicle due to the tail rotor. Body velocities, delayed winds, body angular rates, and main rotor downwash terms are combined to produce tail rotor velocities in body axes. These velocities are transformed to tail rotor shaft axes and the results are used to compute advance ratios. These advance ratios are used to calculate Bailey coefficients. Collective pitch, downwash and inflow are then calculated and used to obtain thrust coefficient and, finally, tail rotor thrust. From the thrust vector, tail rotor forces and moments are computed.

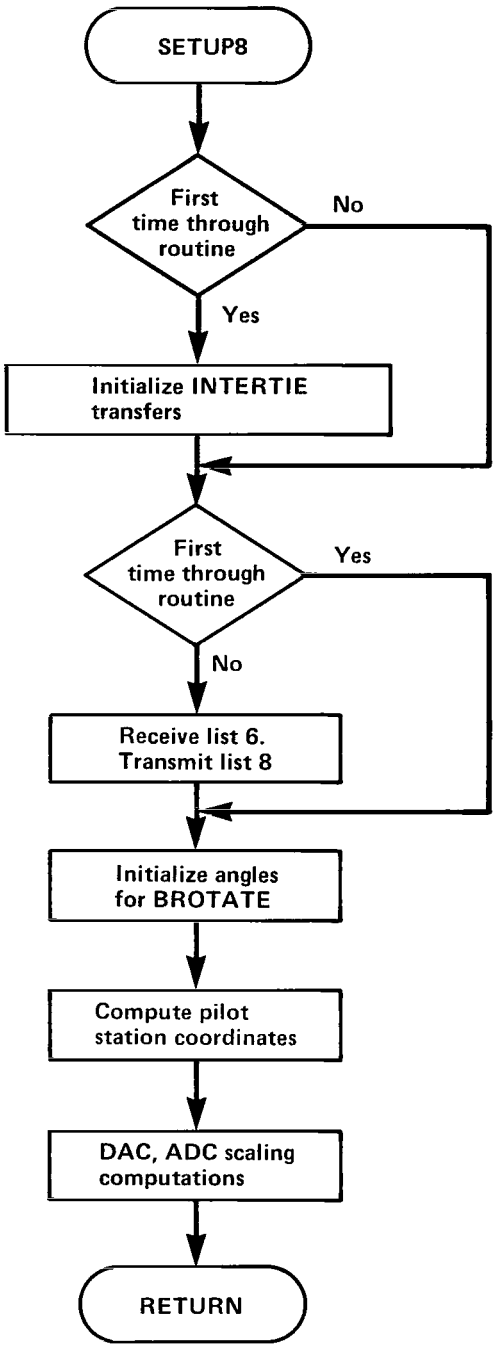
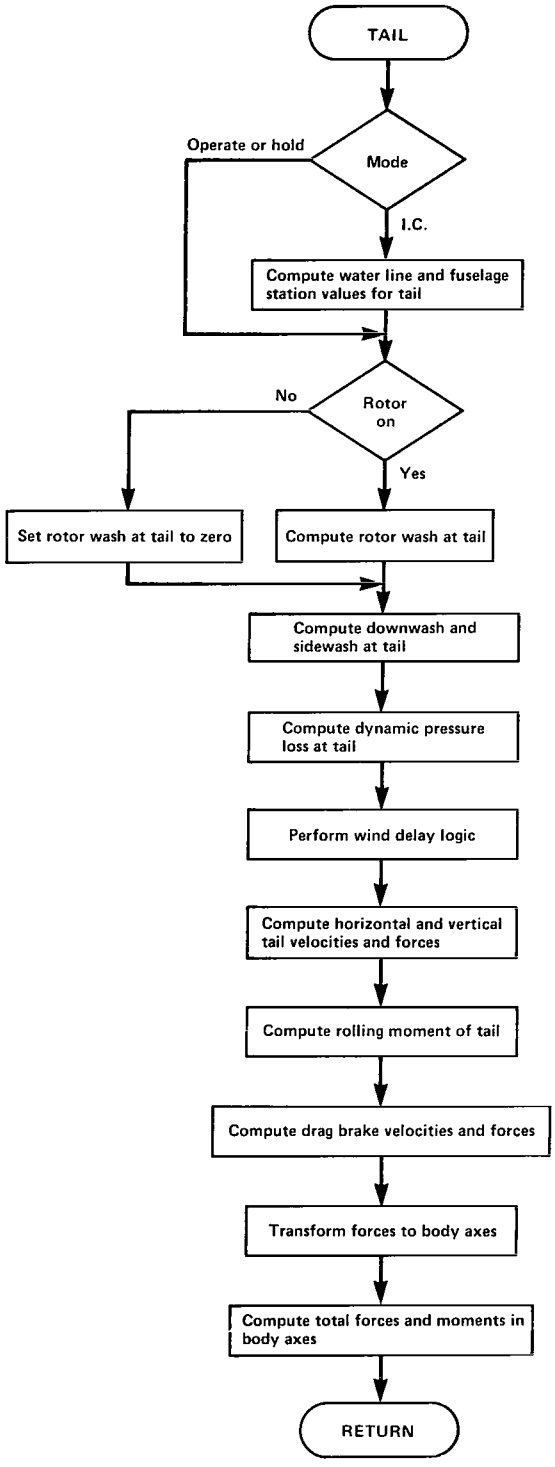


Figure 18.– SETUP8 subroutine.



Note: In fixed wing and compound modes, horizontal tail computations are performed on a 1.6m^2 (17.2 square foot) upper horizontal tail and a 9.1m^2 (98.1 square foot) lower horizontal tail. In helicopter mode, computations are performed only on a 3.3m^2 (35.4 square foot) upper horizontal tail.

Figure 19.– TAIL subroutine.

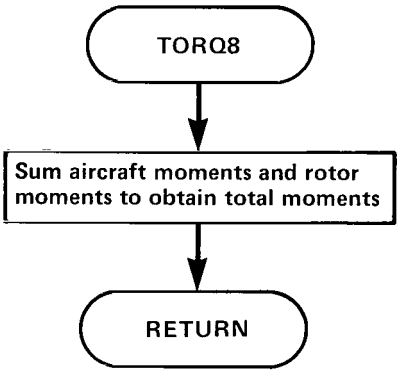


Figure 20.– TORQ8 subroutine.

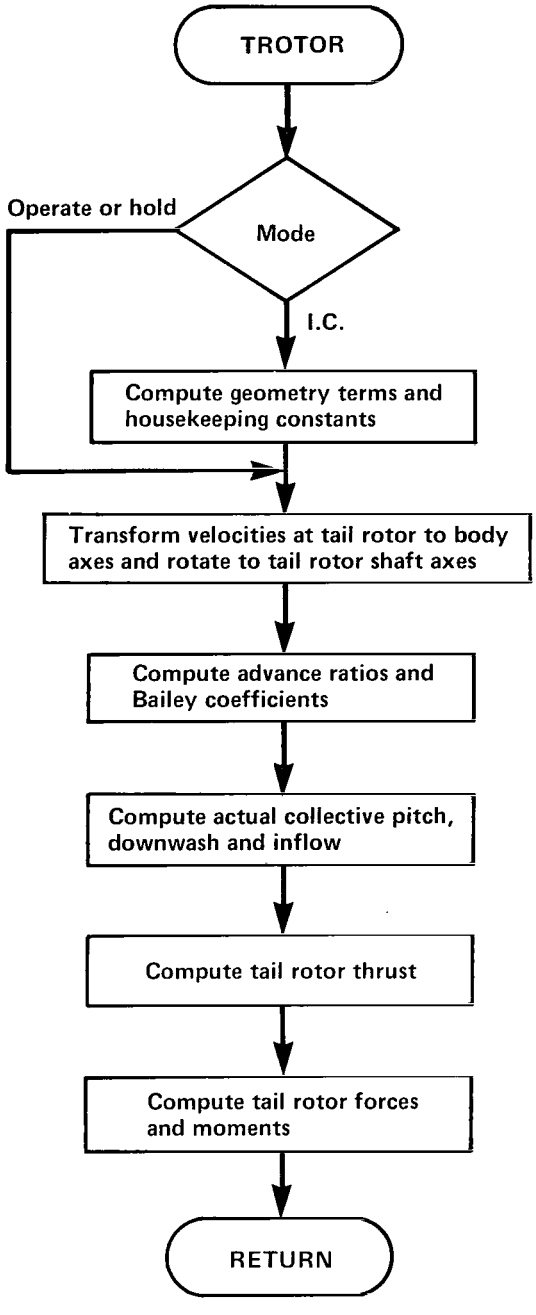


Figure 21.– TROTOR subroutine.

UTIL7

UTIL7 (fig. 22) is the "utility" program for the Sigma 7. In cases where the user requests a print or a strip chart recorder assignment printout, UTIL7 calls in the appropriate overlays and restores the original overlay after the print has been completed. Calls to the dynamic check subroutine are initiated. Special modifications allow for ramp and sine wave inputs to controls, in addition to pulses, steps, and doublets allowed for by the standard dynamic check routine. Dummy lists, used for prints, dynamic checks and mode requests are packed and unpacked. Cycle indicators for the Sigma 7 are computed.

UTIL8

UTIL8 (fig. 23) is the "utility" program for the Sigma 8. Upon command, the D/A scaling program is called. In cases where the user requests a strip chart recorder assignment printout, the recorder printout is called. Automatic strip chart recorder control is made available. Dummy lists, as in UTIL7, are packed and unpacked. Cycle indicators for the Sigma 8 are computed.

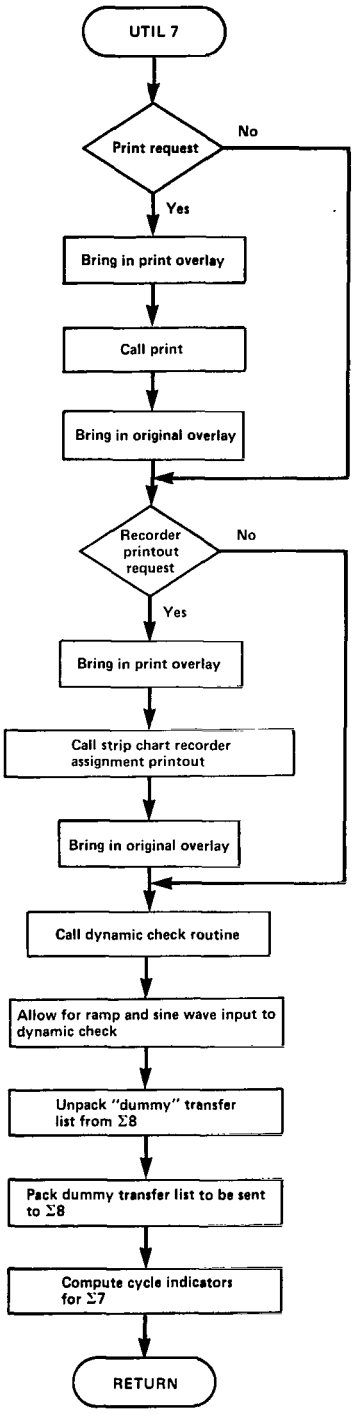


Figure 22.– UTIL7 subroutine.

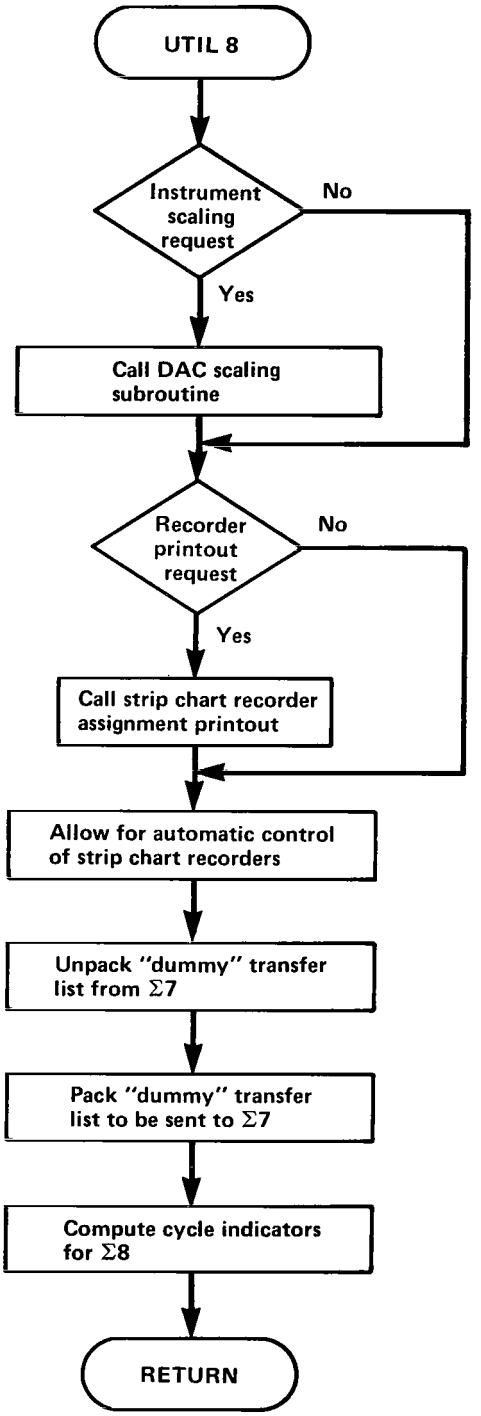


Figure 23.– UTIL8 subroutine.

APPENDIX D

DATA TRANSFER LISTS

List No. 1 – Transferred from Sigma 8 to Sigma 7 in all modes

Item	Definition
PB	Aircraft roll rate
QB	Aircraft pitch rate
RB	Aircraft yaw rate
PBD	Aircraft roll acceleration
QBD	Aircraft pitch acceleration
RBD	Aircraft yaw acceleration
PHI	Aircraft roll angle
THET	Aircraft pitch angle
PSI	Aircraft yaw angle
PHIR	Aircraft roll angle in radians
THETR	Aircraft pitch angle in radians
PSIR	Aircraft yaw angle in radians
WAITIC	Aircraft weight without rotor
TIME	Time since beginning of last operate mode
PBWN	Aircraft roll rate with turbulence
QBWN	Aircraft pitch rate with turbulence
RBWN	Aircraft yaw rate with turbulence
CPLGTRM	Longitudinal CPU trim position
CPLTTRM	Lateral CPU trim position
PT	Total body roll rate in local frame
QT	Total body pitch rate in local frame
RT	Total body yaw rate in local frame
NOROT	Switch indicating "no rotor"
IPDAMP	Roll SAS switch
IQDAMP	Pitch SAS switch
IRDAMP	Yaw SAS switch
IRLSWA	Rotor release switch
IRLSWB	Rotor release switch
ISEHO	Elevator SAS hard-over
ISAHO	Aileron SAS hard-over
ISRHO	Rudder SAS hard-over
ISTRHO	Tail-rotor SAS hard-over
ITRHO	Tail-rotor hard-over
IFLPHO	Flap hard-over
IWNGJM	Wing-incidence jammed
IDRGF	Drag-brake failure
IFLO	Longitudinal force-feel system failure
IFLA	Lateral force-feel system failure

IFDR	Directional force-feel system failure
IJTPFL	Port engine failure
IJTSFL	Starboard engine failure
NOTROTSW	Tail-rotor failure

List No. 2 – Transferred from Sigma 8 to Sigma 7 in all modes.

This list consists of COMMON block ROTOROUT, which is listed in appendix B.

List No. 3 – Transferred from Sigma 7 to Sigma 8 in all modes.

This list consists of COMMON block ACOUT, which is listed in appendix B.

List No. 4 – Transferred from Sigma 8 to Sigma 7 in all modes.

This list consists of an empty block of 15 "dummy" variables.

List No. 5 – Transferred from Sigma 7 to Sigma 8 in all modes.

This list consists of an empty block of 15 "dummy" variables.

List No. 6 – Transferred from Sigma 7 to Sigma 8 in I.C. mode only.

Item	Definition
VEQIC	Initial equivalent air speed in knots
ICONFIG	Configuration indicator
LAGLIM	Rotor lag angle limit indicator
FSCG	Fuselage station of center of gravity
WLCG	Water line of center of gravity
WEIGHT	Total weight of aircraft with rotor
XMC1	Moment of inertia coefficient
XMC2	Moment of inertia coefficient
.	. . . .
.	. . . .
.	. . . .
XMC10	Moment of inertia coefficient
PHIIC	Initial roll angle
THETIC	Initial pitch angle
PSIIC	Initial yaw angle
PBIC	Initial roll rate
QBIC	Initial pitch rate
RBIC	Initial yaw rate

ITPROG	Trim progress indicator
ITRIM	Trim initiated
ITRLIM	Control surface limit reached during trim
XIC	Initial aircraft x position w/r/t runway
YIC	Initial aircraft y position w/r/t runway
HIC	Initial aircraft height w/r/t runway
XMASS	Mass of aircraft
GKCPULG	Longitudinal CPU gain
GKCPULT	Lateral CPU gain
XATRM	Lateral stick trim position
XBTRM	Longitudinal stick trim position
THOL	Tail rotor collective lower limit
THOU	Tail rotor collective upper limit

List No. 7 – Transferred from Sigma 7 to Sigma 8 in all modes.

Item	Definition
FTX	Total aircraft x direction force
FTY	Total aircraft y direction force
FTZ	Total aircraft z direction force
VND	Northward acceleration over earth's surface
VED	Eastward acceleration over earth's surface
VDD	Downward acceleration toward earth's surface
XPR	Distance of pilot down runway
YPR	Distance of pilot to the right of runway
HPR	Distance of pilot above runway
T11	Transformation elements, local to body
.	. . . . .
.	. . . . .
.	. . . . .
T33	Transformation elements, local to body
VEQ	Equivalent air speed
ALT	Altitude of aircraft
ALTD	Altitude rate of change
ALFA	Angle of attack
BETA	Sideslip angle
RPMJETP	Port engine rpm
RPMJETS	Starboard engine rpm
ELEV	Elevator deflection
RUD	Rudder deflection
WINC	Wing incidence
FLAP	Flap deflection
CPUDIR	Directional CPU position
ELVTRM	Elevator trim position
DPEDAL	Pedal deflection
AILTRM	Aileron trim position

RHO	Air density
SOUND	Speed of sound
TAL	Roll moment due to fixed-wing aircraft
TAM	Pitch moment due to fixed-wing aircraft
TAN	Yaw moment due to fixed-wing aircraft
PLB	Roll rate of L frame due to body translation
QLB	Pitch rate of L frame due to body translation
RLB	Yaw rate of L frame due to body translation
PTURB	Roll rate noise
QTURB	Pitch rate noise
RTURB	Yaw rate noise
NCYC1	Cycle indicators
.	. .
.	. .
NCYC4	Cycle indicators
QWF	Dynamic pressure at wing-fuselage

List No. 8 – Transferred from Sigma 8 to Sigma 7 in I.C. only.

Item	Definition
A1S	Tail rotor lateral cyclic
B1S	Tail rotor longitudinal cyclic
CA1S	Lateral CPU gain
CB1S	Longitudinal CPU gain
TKB	SAS gain
RKB	SAS gain
BLRK	SAS gain
TKA1	SAS gain
RKA1	SAS gain
A1LRK	SAS gain
RKYA1	SAS gain
WOB	SAS gain

APPENDIX E

DATA ACQUISITION

Data acquisition for the RSRA simulation consists of two basic types: static and dynamic data. Static data are available in three formats, all printed on an electrostatic printer-plotter. For the RSRA, the dual processor structure necessitated the use of two printer-plotters, one for variables on the Sigma 7 and one for variables on the Sigma 8.

The first type of format allows any contiguous block of variables to be printed. This block may be of any length. The system requires absolute starting and ending addresses, which can be obtained from the load map. The value in each address is printed out with its associated symbolic name. An example of this type of printout is found in figure 24 (a).

The second format allows the values of any predetermined set of variables to be printed. The variables may be contained in a file on the computer's rapid access disk (RAD) or on punched cards. Thus, a non-contiguous set of variables may be printed out as shown in figure 24(b). The above two printouts are formatted and controlled by the real-time debug system (CASPRE).

The third format is user-requested and is provided by the simulation engineer. Control for the printout resides in a user subroutine called PRINT, which is called when a certain discrete is activated, in most cases by pushing a button on the control panel. The format for this printout is determined by the user. An example of this type of user-requested printout may be found in figure 24(c).

Dynamic data for RSRA was available in the form of strip chart recordings. Four strip chart recorders, each with eight multiplexed channels, and two with seven two-way discrete indicators were available, allowing simultaneous real-time or time-scaled recording of 64 variables and 28 discrete events. An example of the multiplexed data from one strip chart recorder can be found in figure 25.

In addition to static and dynamic data related to flight test regimes for the RSRA aircraft, a routine was developed in conjunction with the simulation to provide printout of strip chart recorder assignments upon activation of a discrete value (which was assigned to a panel pushbutton). The printout provides a heading, a list of each channel number and under each channel number, the following information for both the long side and short side of the multiplexed data: variable name, value at left side of channel, value at middle of channel, value at right side of channel, scale, and bias. The information on the printout is spaced in such a way that the page can be cut, and laid in a space available at the bottom of the strip chart recorder, to provide immediate information to the researcher on what data is being presented. An example of the entire strip chart recorder printout is given in figure 26 and a sample of the information as it is used in conjunction with a strip chart recording is found in figure 27. The strip chart recording in this figure is shown in another form available, in which either short or long sides of the multiplexed data can be shown individually by changing certain values on the CRT input keyboard.

```
****CASPRE DUMP****                    12:52   04/15/75

    A = 0897E   Z = 0899E              .PSD =  00000000    00000000

.0 = 0000/000E7                              .1 = 0602/00120
.2 = 0200/0002F                              .3 = 0000/0002F
.4 = 0000/00000                              .5 = 0000/008F0
.6 = 0000/000E7                              .7 = 0602/00120
.8 = 0200/00035                              .9 = 0000/00035
.10 = 0000/00000                             .11 = 0000/008F4
.12 = 0000/000E7                             .13 = 0602/00120
.14 = 0200/0003B                             .15 = 0000/0003B
.0 = +1.189086E-82                           .1 = +2.266407E-72
.2 = +6.193541E-81                           .3 = +2.419352E-83
.4 = +0.000000E+00                           .5 = +1.177761E-81
.6 = +1.189086E-82                           .7 = +2.266407E-72
.8 = +6.984206E-81                           .9 = +2.728206E-83
.10 = +0.000000E+00                          .11 = +1.179820E-81
.12 = +1.189086E-82                          .13 = +2.266407E-72
.14 = +7.774871E-81                          .15 = +3.037059E-83


ENGINE = 6A60/9SETUP0                        ENGINE+1 = 35F0/XMVSUM+3
ENGINE+2 = 2290/00001                        ENGINE+3 = 3190/IRS
ENGINE+4 = 6930/ENGINE+12                    ENGINE+5 = 2280/00000
ENGINE+6 = 3580/RPMJETS                      ENGINE+7 = 3580/RPMJETP
ENGINE+8 = 3580/TJTSUM                       ENGINE+9 = 3580/TJTS
ENGINE+A = 3580/TJTP                         ENGINE+B = 3580/TNJT
ENGINE+C = 3580/TMJT                         ENGINE+D = 3580/TLJT
ENGINE+E = 3580/ZJT                          ENGINE+F = 3580/YJT
ENGINE+10 = 3580/XJT                         ENGINE+11 = 6800/ENGINE+1C9
ENGINE+12 = 3290/IA                          ENGINE+13 = 6920/ENGINE+49
ENGINE+14 = 6830/ENGINE+1C9                  ENGINE+15 = 3280/FSCG
ENGINE+16 = 3C80/FSJT                        ENGINE+17 = 3F80/ENGINE+1CB
ENGINE+18 = 3580/XLLJT                       ENGINE+19 = 3280/WLCG
ENGINE+1A = 3C80/WLJT                        ENGINE+1B = 3F80/ENGINE+1CB
ENGINE+1C = 3580/HJT                         ENGINE+1D = 3280/BLJT
ENGINE+1E = 3F80/ENGINE+1CB                  ENGINE+1F = 3580/BJT
ENGINE+20 = 3280/XIJT
```

(a) CASPRE dump.

Figure 24.– RSRA simulation static data.

```
ZT = +8.822200E+01
     ***  PRINTOUT TO CHECK RSRA TRIM POINTS.
   VEQ = +6.004344E+01
   PHI = -1.078842E+00
   THET = -1.357849E+00
   PSI = +8.999997E+01
   ALFA = -1.357607E+00
   BETA = +2.556351E-02
   ALFWF = -1.357608E+00
   BETAWF = +0.000000E+00
   VXB = +1.019810E+02
   VYB = +0.000000E+00
   VZB = -2.416865E+00
   RHO = +2.345943E-03
   SOUND = +1.114433E+03
   PB = +0.000000E+00
   QB = +0.000000E+00
   RB = +0.000000E+00
   PBD = +2.311353E-03
   QBD = +4.475711E-05
   RBD = +2.624614E-04
   VXBDOT = +8.035835E-03
   VYBDOT = +9.734362E-03
   VZBDOT = -2.479125E-03
   FTX = -4.039485E+02
   FTY = +3.341785E+02
   FTZ = -1.729409E+04
   TTL = +1.873511E+01
   TTM = +4.365967E+00
   TTN = +1.651563E+01
   XMR = -1.055664E+02
   YMR = -2.981445E+02
   ZMR = -1.739751E+04
   RML = -2.393411E+03
   RMM = -1.978800E+03
   RMN = +2.361561E+04
   XWF = -2.560740E+02
   YWF = +0.000000E+00
   ZWF = +1.518996E+01
   TLWF = -1.220134E+00
   TMWF = -6.452480E+02
   TNWF = +2.891618E-02
   XT = -1.976254E+01
   YT = +0.000000E+00
   ZT = +8.822200E+01
   TLT = +0.000000E+00
   TMT = +2.714470E+03
   TNT = +0.000000E+00
   XTR = -2.254582E+01
   YTR = +6.323230E+02
   ZTR = -1.776070E-04
   TRL = +2.413366E+03
   TRM = -8.605649E+01
   TRM = -8.605649E+01
   TRN = -2.359912E+04
   XJT = +0.000000E+00
   YJT = +0.000000E+00
   ZJT = +0.000000E+00
   TLJT = +0.000000E+00
   TMJT = +0.000000E+00
   TNJT = +0.000000E+00
   CHI = +8.085475E+01
   DOWNW = +1.903157E-02
   A1S = -1.388228E+00
   B1S = +9.327135E-01
   THETA0 = +1.212355E+01
   WINC = +0.000000E+00
   XA = +4.819153E+01
   XB = +4.974121E+01
   DPEDAL = +7.463629E+01
   COLSTK = +3.614719E+01
   CPULAT = +1.000000E+02
   CPULON = +1.000000E+02
   CPUDIR = +1.000000E+02
```

(b) CASPRE printout.

Figure 24.– Continued.

NASA-AMES RESEARCH CENTER

ROTOR SYSTEM RESEARCH AIRCRAFT

TIME = .00 DATE 15:59 FEB 11,'75

VEQ	.60043E 02	KTS	ALT	.49145E 03	FT	PHIB	-.15171E 01	DEG
ALFWF	-.32462E 01	DEG	THET	.38303E 01	DEG	GAMV	.00000E 00	DEG
AIL	-.36219E-07	DEG	ELEV	-.58731E 01	DEG	RUD	.00000E 00	DEG
FLAP	.00000E 00	DEG	IWNG	.00000E 00	DEG	DRAG	.00000E 00	DEG
DWSHMR	.19066E-01		LAMBMR	-.14134E-01		CHI	.89979E 02	DEG
A1S	-.12938E 01	DEG	B1S	.55800E 01	DEG	THETA0	.11851E 02	DEG
NBS	.40000E 01		NSS	.30000E 01		OMEGMR	.21290E 02	R/S
BR	.26576E 01	DEG	LAG	-.59105E 01	DEG	LSHARE	.91400E 00	
CDELA	.59605E-07		CDELE	.59605E-07		CTTR	.10000E 01	
CA1S	.10000E 01		CB1S	.10000E 01		OMEGTR	.13040E 03	R/S
THETTR	.56170E 01	DEG	IHT	-.25000E 01	DEG	RPMJET	.00000E 00	PCT
XJET	.00000E 00	LB	MJET	.00000E 00	LB	ZJET	.00000E 00	LB
XMR	.16045E 04	LB	YMR	-.22787E 03	LB	ZMR	-.16818E 05	LB
XWF	-.16835E 03	LB	YWF	.60136E 01	LB	ZWF	.18038E 02	LB
XT	-.24785E 02	LB	YT	.17283E 01	LB	ZT	-.92837E 02	LB
XTR	-.16795E 02	LB	YTR	.64576E 03	LB	ZTR	-.18138E-03	LB
LMR	-.23558E 04	FP	MMR	.38917E 04	FP	NMR	.22519E 05	FP
LWF	.83821E 01	FP	MWF	-.18015E 04	FP	NWF	.31368E 03	FP
LT	-.16569E 03	FP	MT	-.40031E 04	FP	NT	-.71847E 02	FP
LTR	.24593E 04	FP	MTR	.63954E 02	FP	NTR	-.23146E 05	FP
WEIGHT	.18400E 05	LB	FSCG	.31350E 03	IN	WLCG	.23080E 03	IN
IX	.85640E 04	SF2	IY	.96918E 05	SF2	IZ	.90698E 05	SF2
IXZ	.33770E 04	SF2	TOTLWT	.13770E 04	LB	TOTDWT	-.19262E 03	LB
QWF	.12191E 02	P/F2	BETAWF	-.10140E 00	DEG	DRE	.70000E 02	
TKB	.14000E 02	SEC	RKB	.52000E 00	DDS	BLRK	.52000E 00	
TKE	.50000E 01	SEC	RKE	.60000E 00	DDS	ELRK	.15000E 01	
TKA1	.80000E 01	SEC	RKA1	.16000E 00	DDS	A1LRK	.12000E 00	
TKA	.50000E 01	SEC	RKA	.20000E 00	DDS	ALRK	.00000E 00	
TK5T	.10000E 01	SEC	RK8T	.31500E 00	DDS	TLRK5	.00000E 00	
TK5R	.30000E 01	SEC	RK8R	.10000E 01	DDS	RLRK5	.30000E 01	
RKYA1	.00000E 00	DDS	RKYA	.00000E 00	DDS	RKR7T	.00000E 00	DDS
RKR7R	.00000E 00	DDS	WOB	.10000E 00	SEC	WOE	.10000E 00	SEC

OMGRAT	.10000E 01		QBARMR	.22814E 05	FP	XA	.46962E 02	PCT
XB	.63769E 02	PCT	XC	.26570E 02	PCT	XP	.69415E 02	PCT
KASAIL	.20000E 01		AILG	.19000E 01			.00000E 00	
	.00000E 00			.00000E 00			.00000E 00	
FALONG	.29037E-02	P/IN	FBLONG	.24414E-03	PS/I	FCLONG	.36568E-03	PS/D
FALAT	.29037E-02	P/IN	FBLAT	.24414E-03	PS/I	FCLAT	.36568E-03	PS/D
FADIR	.80000E 01	P/IN	FBDIR	.00000E 00	PS/I		.00000E 00	
	.00000E 00			.00000E 00			.00000E 00	
KEG	.23493E 01		CTW	-.25000E 01		KTW1	.18000E 00	
KTW2	.00000E 00		KTW5	.00000E 00		CTFLAP	.00000E 00	
KTFLAP	.00000E 00		KTLIN1	.10000E 01		KTLIN5	.00000E 00	
	.00000E 00			.00000E 00			.00000E 00	

(c) User-requested printout.

Figure 24.– Concluded.

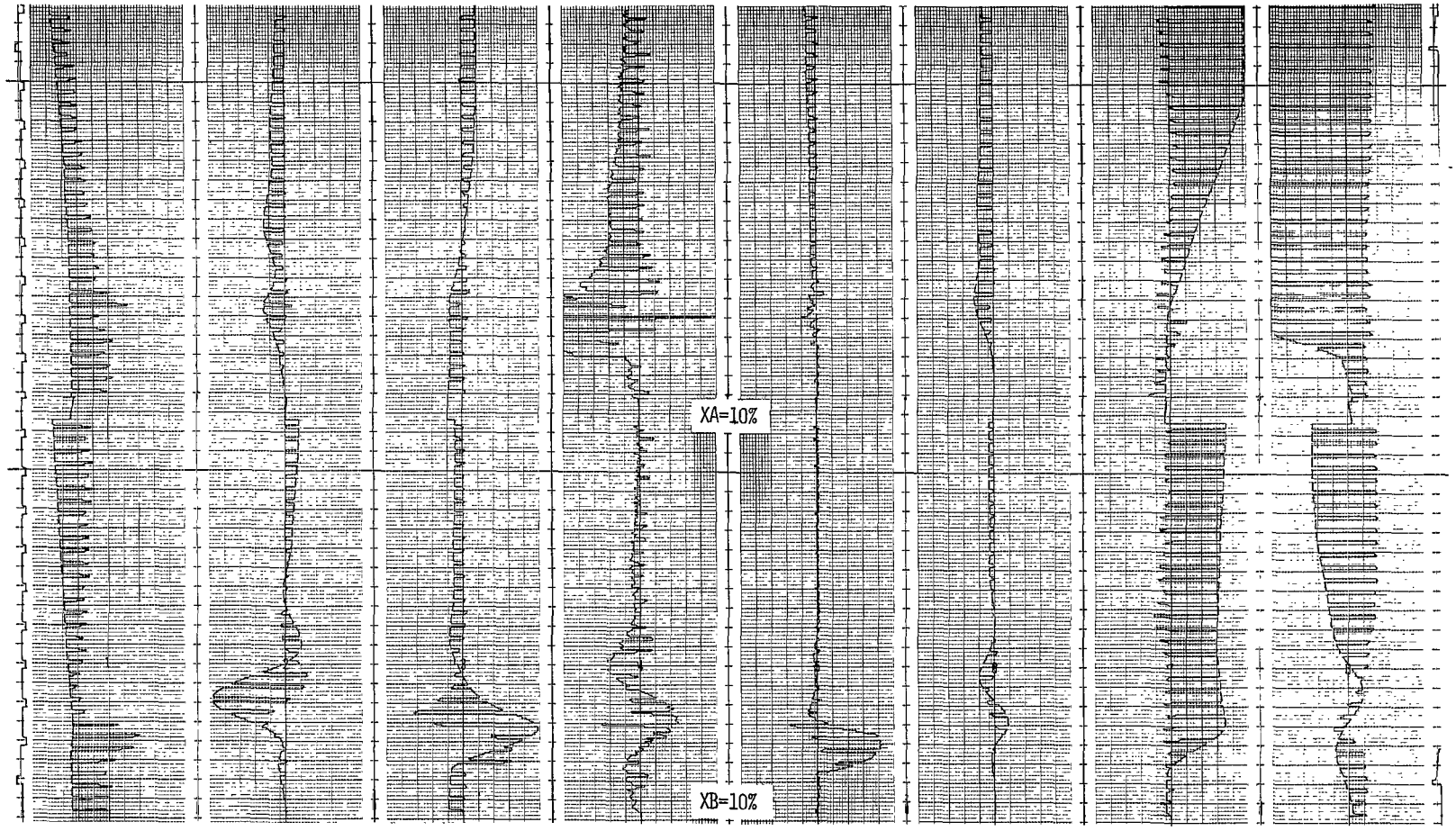


Figure 25.– RSRA dynamic data multiplexed strip chart recording.

THESE ARE THE RECORDER ASSIGNMENTS FOR PROJECT RSRA 15:41 FEB 11,'75

RECORDER NUMBER 1

CHANNEL NO. 1 (0)	CHANNEL NO. 2 (1)	CHANNEL NO. 3 (2)	CHANNEL NO. 4 (3)
LONG SIDE: ACO	LONG SIDE: VYB	LONG SIDE: VZB	LONG SIDE: PB
LEFT = .5000E 03	LEFT = .2500E 02	LEFT = .5000E 02	LEFT = .4363E 00
MIDDLE = .2500E 03	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00
RIGHT = .0000E 00	RIGHT = -.2500E 02	RIGHT = -.5000E 02	RIGHT = -.4363E 00
SCALE = .2500E 03	SCALE = .2500E 02	SCALE = .5000E 02	SCALE = .4363E 00
BIAS = -.1000E 01	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00
SHORT SIDE: VXBDOT	SHORT SIDE: VYBDOT	SHORT SIDE: VZBDOT	SHORT SIDE: PBD
LEFT = .2500E 02	LEFT = .1250E 03	LEFT = .1250E 03	LEFT = .2500E 01
MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00
RIGHT = -.2500E 02	RIGHT = -.1250E 03	RIGHT = -.1250E 03	RIGHT = -.2500E 01
SCALE = .2500E 02	SCALE = .1250E 03	SCALE = .1250E 03	SCALE = .2500E 01
BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00

CHANNEL NO. 5 (4)	CHANNEL NO. 6 (5)	CHANNEL NO. 7 (6)	CHANNEL NO. 8 (7)
LONG SIDE: QB	LONG SIDE: RB	LONG SIDE: THET	LONG SIDE: A
LEFT = .4363E 00	LEFT = .4363E 00	LEFT = .2500E 02	LEFT = .2500E 02
MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00
RIGHT = -.4363E 00	RIGHT = -.4363E 00	RIGHT = -.2500E 02	RIGHT = -.2500E 02
SCALE = .4363E 00	SCALE = .4363E 00	SCALE = .2500E 02	SCALE = .2500E 02
BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00
SHORT SIDE: QBD	SHORT SIDE: RBD	SHORT SIDE: BR	SHORT SIDE: XLAG
LEFT = .2500E 01	LEFT = .2500E 01	LEFT = .4363E 00	LEFT = .4363E 00
MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00
RIGHT = -.2500E 01	RIGHT = -.2500E 01	RIGHT = -.4363E 00	RIGHT = -.4363E 00
SCALE = .2500E 01	SCALE = .2500E 01	SCALE = .4363E 00	SCALE = .4363E 00
BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00

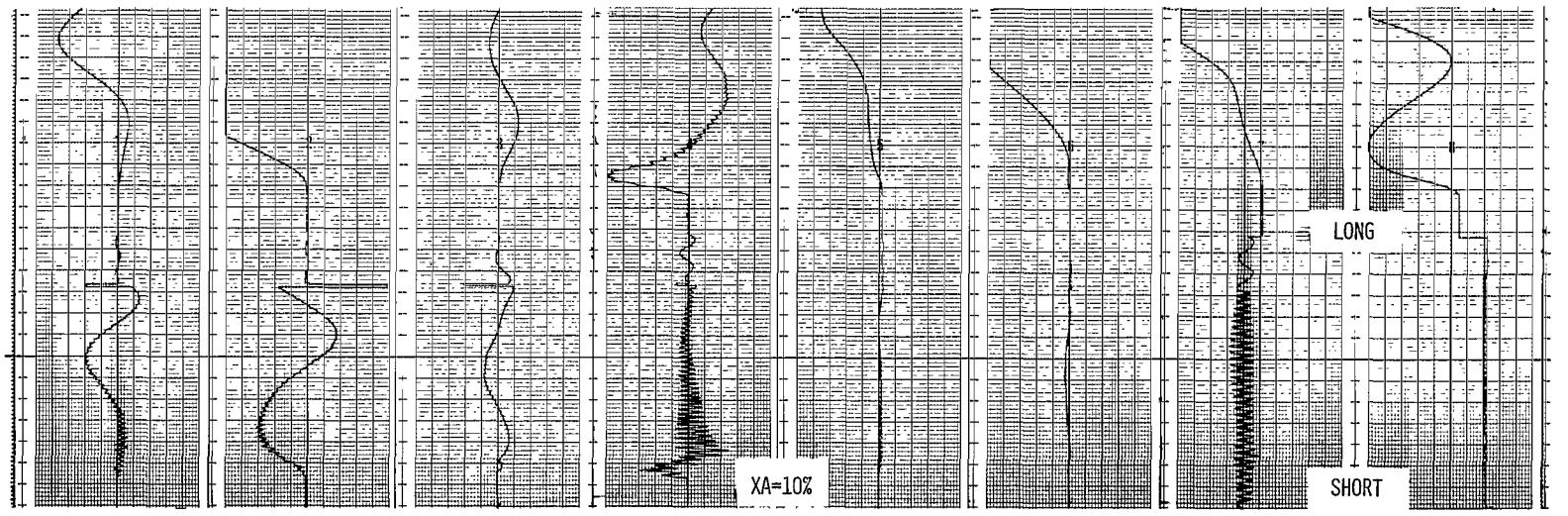
Figure 26.– RSRA dynamic data recorder printout.

RECORDER NUMBER 2

CHANNEL NO. 1 (8)	CHANNEL NO. 2 (9)	CHANNEL NO. 3 (10)	CHANNEL NO. 4 (11)
LONG SIDE: PSI	LONG SIDE: ALFWF	LONG SIDE: BETAWF	LONG SIDE: BLANK
LEFT = .1800E 03	LEFT = .2500E 02	LEFT = .2500E 02	LEFT = .9999E 04
MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00
RIGHT = -.1800E 03	RIGHT = -.2500E 02	RIGHT = -.2500E 02	RIGHT = -.9999E 04
SCALE = .1800E 03	SCALE = .2500E 02	SCALE = .2500E 02	SCALE = .9999E 04
BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00
SHORT SIDE: XA	SHORT SIDE: XB	SHORT SIDE: DPEDAL	SHORT SIDE: COLSTK
LEFT = .1000E 03	LEFT = .1000E 03	LEFT = .1000E 03	LEFT = .1000E 03
MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00
RIGHT = -.1000E 03	RIGHT = -.1000E 03	RIGHT = -.1000E 03	RIGHT = -.1000E 03
SCALE = .1000E 03	SCALE = .1000E 03	SCALE = .1000E 03	SCALE = .1000E 03
BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00

CHANNEL NO. 5 (12)	CHANNEL NO. 6 (13)	CHANNEL NO. 7 (14)	CHANNEL NO. 8 (15)
LONG SIDE: VEQ	LONG SIDE: BLANK	LONG SIDE: BLANK	LONG SIDE: BLANK
LEFT = .5000E 03	LEFT = .1000E 07	LEFT = .9999E 04	LEFT = .9999E 04
MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00
RIGHT = -.5000E 03	RIGHT = -.1000E 07	RIGHT = -.9999E 04	RIGHT = -.9999E 04
SCALE = .5000E 03	SCALE = .1000E 07	SCALE = .9999E 04	SCALE = .9999E 04
BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00
SHORT SIDE: XDRAG	SHORT SIDE: XWNG	SHORT SIDE: XFLAP	SHORT SIDE: XRPMJTP
LEFT = .1000E 03	LEFT = .1000E 03	LEFT = .1000E 03	LEFT = .1000E 03
MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00
RIGHT = -.1000E 03	RIGHT = -.1000E 03	RIGHT = -.1000E 03	RIGHT = -.1000E 03
SCALE = .1000E 03	SCALE = .1000E 03	SCALE = .1000E 03	SCALE = .1000E 03
BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00

Figure 26.– Concluded.



CHANNEL NO. 1 (0)	CHANNEL NO. 2 (1)	CHANNEL NO. 3 (2)	CHANNEL NO. 4 (3)	CHANNEL NO. 5 (4)	CHANNEL NO. 6 (5)	CHANNEL NO. 7 (6)	CHANNEL NO. 8 (7)
LONG SIDE; ACD	LONG SIDE: VYB	LONG SIDE: VZB	LONG SIDE: PB	LONG SIDE: QB	LONG SIDE: RB	LONG SIDE: THET	LONG SIDE: A
LEFT = .5000E 03	LEFT = .2500E 02	LEFT = .5000E 02	LEFT = .4363E 00	LEFT = .4363E 00	LEFT = .4363E 00	LEFT = .2500E 02	LEFT = .2500E 02
MIDDLE = .2500E 03	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00
RIGHT = .0000E 00	RIGHT = -.2500E 02	RIGHT = -.5000E 02	RIGHT = -.4363E 00	RIGHT = -.4363E 00	RIGHT = -.4363E 00	RIGHT = -.2500E 02	RIGHT = -.2500E 02
SCALE = .2500E 03	SCALE = .2500E 02	SCALE = .5000E 02	SCALE = .4363E 00	SCALE = .4363E 00	SCALE = .4363E 00	SCALE = .2500E 02	SCALE = .2500E 02
BIAS = -.1000E 01	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00
SHORT SIDE: VXBDOT	SHORT SIDE: VYBDOT	SHORT SIDE: VZBDOT	SHORT SIDE: PBD	SHORT SIDE: QBD	SHORT SIDE: RBD	SHORT SIDE: BR	SHORT SIDE: XLAG
LEFT = .2500E 02	LEFT = .1250E 03	LEFT = .1250E 03	LEFT = .2500E 01	LEFT = .2500E 01	LEFT = .2500E 01	LEFT = .4363E 00	LEFT = .4363E 00
MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00	MIDDLE = .0000E 00
RIGHT = -.2500E 02	RIGHT = -.1250E 03	RIGHT = -.1250E 03	RIGHT = -.2500E 01	RIGHT = -.2500E 01	RIGHT = -.2500E 01	RIGHT = -.4363E 00	RIGHT = -.4363E 00
SCALE = .2500E 02	SCALE = .1250E 03	SCALE = .1250E 03	SCALE = .2500E 01	SCALE = .2500E 01	SCALE = .2500E 01	SCALE = .4363E 00	SCALE = .4363E 00
BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00	BIAS = .0000E 00

Figure 27.– RSRA dynamic data strip chart recording with recorder printout.

REFERENCES

1. Zuccaro, Joseph J.: The Flight Simulator for Advanced Aircraft – A New Aeronautical Research Tool. AIAA Paper 70-359, March 1970.

2. Raby, James S.: NASA-Ames Research Center Hybrid Computer Operating System. Proceedings of the Conference on Hybrid Computation, Munich, West Germany, Aug. 31–Sept. 4, 1970, pp. 672–674.

3. McFarland, Richard E.: A Standard Kinematic Model for Flight Simulation at NASA-Ames. NASA CR-2497, Jan. 1975.

C08 001 C1 U A 770107 S00903DS
DEPT OF THE AIR FORCE
AF WEAPONS LABORATORY
ATTN: TECHNICAL LIBRARY (SUL)
KIRTLAND AFB NM 87117

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons. Also includes conference proceedings with either limited or unlimited distribution.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include final reports of major projects, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION OFFICE

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Washington, D.C. 20546